NASA Technical Memorandum 104738 P.230

Making Intelligent Systems Team Players: Case Studies and Design Issues Volume 2: Fault Management System Cases

Jane T. Malin
Debra L. Schreckenghost
David D. Woods
Scott S. Potter
Leila Johannesen
Matthew Holloway

(NASA-TM-104738) MAKING INTELLIGENT SYSTEMS TEAM PLAYERS: CASE STUDIES AND DESIGN ISSUES. VOLUME 2: FAULT MANAGEMENT SYSTEM CASES (NASA) 230 p CSCL 12A N92-11641 Unclas G3/59 0048389

October 1991

NASA

NASA Technical Memorandum 104738

Making Intelligent Systems Team Players: Case Studies and Design Issues
Volume 2: Fault Management System Cases

Jane T. Malin
Lyndon B. Johnson Space Center
Houston, Texas

Debra L. Schreckenghost
The Mitre Corportation
Houston, Texas

David D. Woods
Scott S. Potter
Leila Johannesen
Matthew Holloway
Cognitive Systems Engineering Laboratory
The Ohio State University

National Aeronautics and Space Administration
Lyndon B. Johnson Space Center
Houston, Texas

October 1991

Abstract

Observations from a case study of intelligent systems are reported as part of a multi-year, interdisciplinary effort to provide guidance and assistance for designers of intelligent systems and their user interfaces. The objective of this case study was to identify preliminary guidance for the design of effective human-computer interaction (HCI) with intelligent fault management systems in aerospace applications. Fifteen intelligent fault management systems within the National Aeronautics and Space Administration (NASA) were studied. Preliminary results based on this case study are documented in Volume 1 of this report, *Making Intelligent Systems Team Players: Case Studies and Design Issues, Volume 1. Human-Computer Interaction Design.*

Keywords: human-computer interaction, user interface, intelligent system, design guidance, development methodology, real-time fault management

Acknowledgements

We would like to thank all the NASA and contractor personnel who took time to demonstrate and explain their systems to us. Their patience and willingness to share their problems and ideas fostered an ideal environment in which to undertake this work. We also appreciate the time expended by these personnel to review the case study reports for accuracy.

Table of Contents

Table of Contents (continued)

Table of Contents (continued)

Section **Page**

Table of Contents (continued)

List of Figures

Figure **Page**

PART I: Case Study Performed by Study Team from Johnson Space Center

List of Figures (continued)

List of Figures (continued)

Figure | Page

PART II: Case Study Performed by Study Team from Ohio State University

List of Tables

Table **Page**

PART I: Case Study Performed by Study Team from Johnson Space Center

Introduction

In 1990, a study was initiated to provide guidance in designing intelligent systems that are effective team members in flight operations support. This study was conducted as part of a Research and Technology Operating Plan (RTOP) for the Artificial Intelligence Division of the Office of Aeronautics, Explorations, and Technology (OAET). This report documents the observations from a case study performed as part of the RTOP to investigate human-computer interaction (HCI) design guidance. The objective of this case study was to identify preliminary guidance for the design of effective HCI with intelligent fault management systems in aerospace applications. The results of the case study should be of interest to intelligent system designers and to researchers in the areas of HCI and human factors, artificial intelligence, and software engineering.

Two study teams participated in this case study. One team was located at the Johnson Space Center (JSC) and consisted of Dr. Jane Malin of the Intelligent System Branch and Debra Schreckenghost of the MITRE Corporation. This team is referred to as the JSC study team. The other team was located at Ohio State University (OSU) and consisted of Dr. David Woods, Scott Potter, Leila Johannesen, and Matthew Holloway. This team is referred to as the OSU study team.

The investigation has been limited to fault management systems within aerospace domains. For this study, a fault management system is a software support system that assists a flight controller in real-time monitoring for fault detection, isolation, and recovery (FDIR). The study was also constrained to human-centered applications, where responsibility for the control of both the monitored process and the intelligent support system lies with the operator. Thus the computer becomes a technical assistant to the operator.

Information about the applications selected for this case study was acquired by interviewing the developers and users of NASA fault management intelligent systems. These interviews were supplemented by relevant documentation and demonstrations of these applications. The information collected included a domain description, intelligent system functionality, and supporting HCI and user interface capabilities. The design techniques and methodologies used were also documented in an attempt to characterize the design process. Information collected during the case study has been used to identify candidate guidelines, interesting examples, key design issues, promising research areas, effective design methodologies, and characteristics of the design process.

This report is the second volume of a two volume set documenting the results of the case study. This volume (*Volume 2. Fault Management System Cases*) describes study team observations about each application in the case study. *Volume 1. Human-Computer Interaction Design* (Malin et al., 1991) presents preliminary HCI design guidance for intelligent system developers based on these study observations.

This report is segmented into two main parts. Part I presents the results of the cases evaluated by the JSC study team. Part II presents the results of the cases evaluated by the OSU study team. This organization was selected to accommodate the different documentation styles of each team. In both parts, each case is described in a separate section, and the same subsection organization is used for each case description. Part I begins with a section summarizing observations and issues from all cases studied at JSC. In Part II, a summary of issues is provided at the end of each case description.

The description of each application in the case study discusses the following topics:

- System, including the monitored process and overall human-computer system
- Intelligent system and its functions
- Human-intelligent system interaction
- User interface capabilities

Additionally, observations are made about the development environment and the methods used to design and develop the system. All points of contact made during the case study are listed in the appendix.

Reference

Malin, J. T., D. L.Schreckenghost, D. D. Woods, S. S. Potter, L. Johannesen, M. Holloway, and K. D. Forbus (September, 1991), *Making Intelligent Systems Team Players: Case Studies and Design Issues, Volume 1. Human-Computer Interaction Design*, NASA Technical Memo, Houston, TX: NASA - Johnson Space Center.

Part I

Case Study Performed by Study Team from Johnson Space Center

Section 1
Introduction

Part I documents observations from a case study of intelligent systems performed by the study team located at the Johnson Space Center (JSC). The case study examined human-intelligent system interaction, user interface capabilities, and the design process used in on-going NASA projects to develop intelligent fault management systems. Applications evaluated by the JSC study team include:

- Real-Time Data Systems (RTDS) Applications at Johnson Space Center
  - Space Shuttle Guidance, Navigation and Control (GNC) Intelligent Systems
  - Space Shuttle Instrumentation and Communications Officer (INCO) Expert System Project (IESP)
  - Space Shuttle KU Band Self Test Expert System
  - Space Shuttle DATA COMM Expert System
  - Space Shuttle Payload Deployment and Retrieval System (PDRS) Decision Support System (DESSY)
- X-29 Remotely Augmented Vehicle Expert System (RAVES) at Ames Research Center's (ARC) Dryden Flight Research Facility
- Military Aircraft Short Take Off and Landing (STOL) Real-Time Interactive Monitoring Systems (RTIMES) at Edwards Air Force Base (EAFB)
- Space Shuttle Onboard Navigation (ONAV) Expert System at Johnson Space Center
- Space Shuttle Rendezvous Expert System (REX) at Johnson Space Center
- Space Station Operations Management System (OMS) Prototypes at Johnson Space Center

In section 2, observations from the case study are summarized. This summary provides an overview of all applications surveyed. Subsequent sections document observations about each of the intelligent systems studied.

Development of most of the intelligent systems in the case study is on-going. All descriptions of applications refer to the system at the time that system developers and users were interviewed and may not reflect the current system.

Section 2
Summary of Observations

This section summarizes the case observations made by the JSC study team. The primary method for collecting information about these applications was interview of system developers and users. In many cases, these interviews were accompanied by a demonstration of the application. Where possible, available documentation (e.g., briefings, requirements documents, papers, etc.) was used to supplement the information gained in the interview. Phone calls and additional visits were used to clarify unclear observations and correct inaccuracies. The final interview reports were reviewed by the individuals that had provided the information to ensure that the application was accurately described. See the appendix for a list of contacts made during the interviews.

2.1 System Description

Description of Flight Support Environment

The Mission Control Center (MCC) for Space Shuttle operations is the most mature flight support environment encountered in the case study and is representative of a "typical" flight control center within NASA. The subsequent description is primarily based on the MCC. Ground flight support for the Space Station will be similar to Space Shuttle flight support, although the longer-term mission will necessitate greater use of automation to alleviate the human workload. Where differences are expected for Space Station, they are described.

In addition to the astronauts, a crew of ground-based flight controllers at the Johnson Space Center provide continuous mission support during a Space Shuttle flight. Flight control positions are categorized by primary vehicle systems and ground-based support systems. These positions are organized in a control hierarchy with communication between levels. The Flight Director, responsible for coordinating the overall mission, is located at the top of this hierarchy. The next layer consists of controllers located in the Flight Control Room (FCR), which determine and communicate status, configuration, and recommendations concerning the primary systems to the Flight Director. The controllers in the FCR are supported by the third layer, which consists of controllers located at consoles in the Multi-Purpose Support Rooms (MPSRs). These controllers monitor specific subsystems of the primary systems and provide status, configuration, and recommendations during anomalies to the FCR operators. Additionally, design engineering support is provided by system design engineers monitoring real-time telemetry in the Mission Evaluation Room (MER). The Spacecraft Analysis (SPAN) flight support personnel provide an interface between personnel in the FCR and the MER. The hierarchy for Space Shuttle is summarized in figure 2-1.

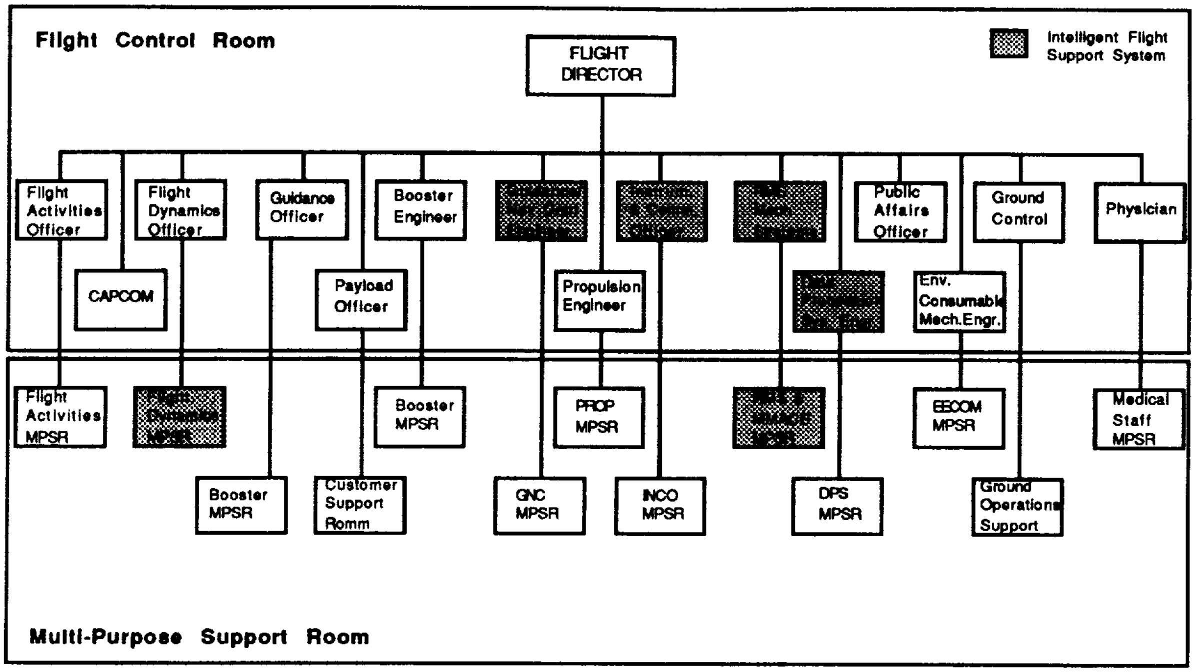


Figure 2-1. Space Shuttle Flight Control Positions (Rasmussen et al., 1990)

This hierarchical support structure was devised to limit communication between layers to significant information only, thus minimizing overload of operators at the coordination level. Timely, coordinated operations is achieved within this structure by providing a well-specified, terse language for communication and by extensive pre-flight planning and documentation of procedures for both nominal and off-nominal situations. Such support requires a significant amount of training to learn both the language of the domain and to be able to perform the detailed, manual procedures. For Space Shuttle, successful flight support relies heavily on paper documentation. A significant difference expected with Space Station is the availability of electronic versions of such paper documentation.

Information used for ground-based flight control is communicated across three types of interfaces: human-to-human, human-to-computer, computer-to-computer. Associated with each interface are established means/ways to communicate information across the interface. For Space Shuttle, communication between humans is accomplished using the voice communication loops, discussion in-person, and various forms of printed material (e.g., event logs, display hardcopies, flight documents). For Space Station, electronic communication (e.g., E-mail) is likely to be added as another form of communication.

Space Shuttle operators communicate with both the Mission Operations Computer (MOC) and stand-alone workstations. The MOC displays telemetry data downlisted from the Space Shuttle to the ground operators using screen displays (tabular and limited graphical), light panels indicating parameter states, and strip chart data plots. Humans interact with the ground flight computers using computer terminals called Manual Entry Devices (MEDs) or using Push Button Indicators (PBIs) on the control panel. Valid inputs include orbiter uplink parameters, parameters for configuring ground support systems, request for display hardcopy, and parameters for configuring ground flight consoles to support a specific mission. Ground operators interact with stand-alone workstations to perform data conversions and computations not available from ground flight computers. A major task of these stand-alone workstations is to provide off-line support tools, such as simulation for the verification of maneuvers and trajectories and for the certification of contingency corrective procedures.

Communication between orbiter computers and the ground-based computers is accomplished using communication link via Tracking and Data Relay Satellite System (TDRSS) or ground-based radar stations. Space Shuttle telemetry data are downlisted from the General Purpose Computers (GPC) cyclically at 1 second intervals. Parameters and commands for Space Shuttle systems are uplinked from the MOC as needed (event-driven).

Ground-based flight operations as described above are subject to limitations which can affect the ability of operators to effectively perform their job. Manually intensive procedures executed in a time-critical environment can lead to controller overload. During high stress situations, humans tend to focus on portions of the problem, often ignoring other important aspects of the situation. To avoid loss of critical information, it is necessary to provide extra manpower for redundant monitoring. This extra manpower adds cost to mission support. The use of intelligent system support has been investigated to alleviate some of these difficulties.

Certain aspects of the current flight support environment complicate the use of intelligent systems for ground-based mission support. Most of the constraints of the current system are due to the baseline of old technology with centralized processing and significant memory constraints. In such an environment, the focus was on having real-time data available, regardless of the format. Current user interfaces, primarily tabulated text that stress maximum usage of display space (see figure 2-2), are difficult to learn and use. Controllers must be highly trained to use these data formats to form accurate assessments quickly. User acceptance of an intelligent system in this environment will depend on how well the human-computer interface integrates with the existing support environment. There is significant reliance on additional information not available or accessible in an electronic format. Much critical information is currently only available from the voice loop or on paper. As technology is upgraded in the control centers, such information may become available electronically (e.g., procedures documents may be replaced by on-line electronic procedures in the future). Thus, intelligent systems built for the current support environment should not preclude evolution from manual input of data to electronic sources for data, possibly from other ground-based support positions.

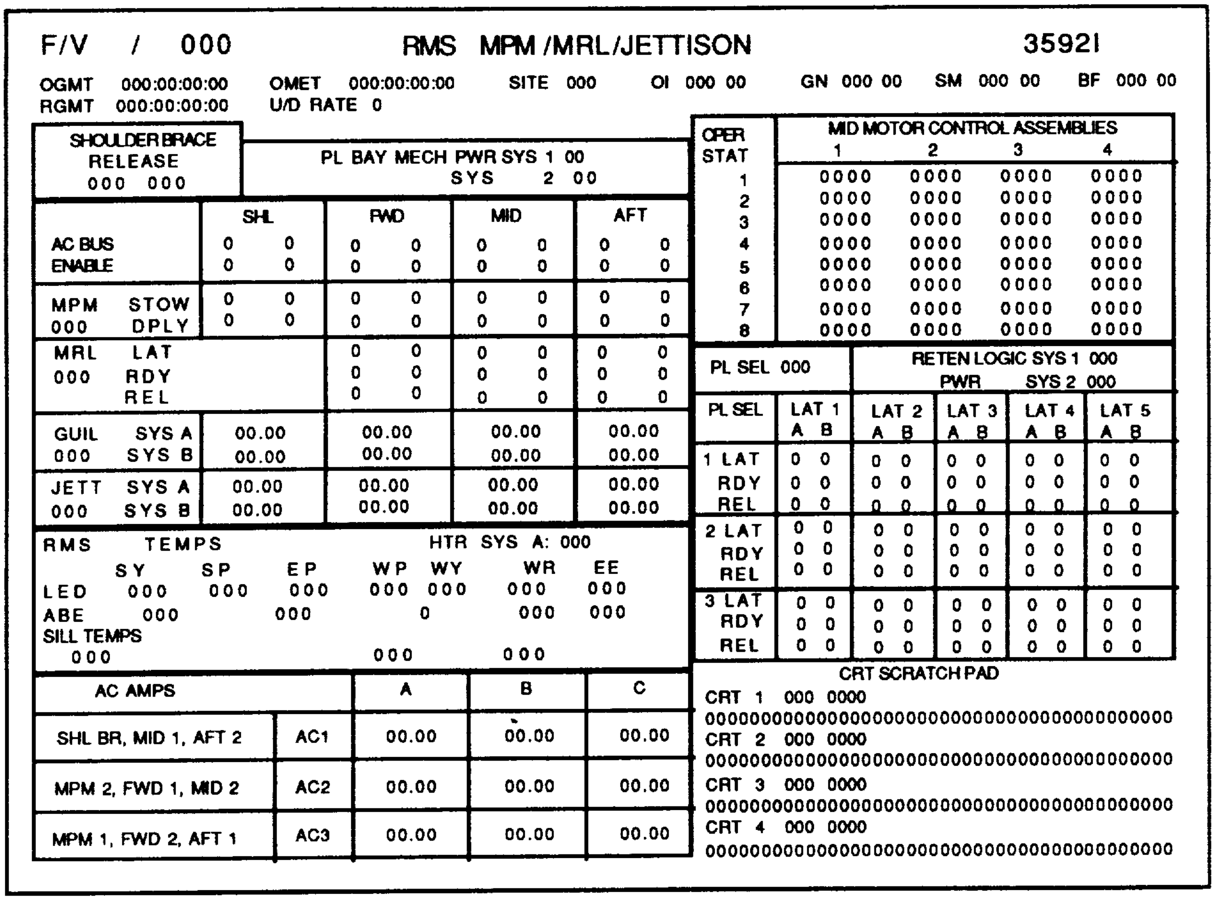


Figure 2-2. Example of Displays Currently Used in Space Shuttle MCC (JSC, 1983)

The Mission Control Center Upgrade (MCCU) is an ongoing project to upgrade the hardware and support software in the Space Shuttle MCC at JSC to reflect recent technology developments. Essential aspects of this upgrade include a change from centralized processing of flight data to distributed processing. The current configuration of main frame computation and display generation (with display at multiple dumb terminals) will be enhanced by multiple workstations connected via Local Area Network (LAN). This upgrade will increase both the processing and memory capacity of the MCC and provide the potential for new, customized processing and display capability for MCC flight controllers. In anticipation of this increased capability, upgrades of the flight controller interfaces and increased automation of flight control operations are currently being prototyped (e.g., Real-Time Data System described in the next section).

Preliminary operational concepts for the Space Station Control Center (SSCC) indicate a distributed processing environment similar to MCCU, with UNIX® workstations connected via LAN.

® - UNIX is a trademark of American Telephone and Telegraph Bell Laboratories

Development and Testing Environments

Three laboratories were responsible for developing the Space Shuttle applications reviewed in this case study:

- Real Time Data Systems (RTDS) Lab in the JSC Mission Operations Directorate (MOD)
- Artificial Intelligence (AI) Lab in the JSC Information Systems Directorate (ISD)
- Intelligent System Branch (ISB) Lab in the JSC Engineering Directorate (ED)

The Space Station OMS Prototypes were developed for use in the Data Management System (DMS Test Bed) of the JSC Engineering Directorate. RAVES was developed for operational use in the Remotely Augmented Vehicle (RAV) Lab of Dryden Flight Research Facility's (DFRF) Integrated Test Facility (ITF). RTIMES was developed for use in the Edwards Air Force Base (EAFB) Short Take Off and Landing (STOL) Maneuver Technology Demonstrator. The salient features of each of these facilities are outlined in table 2-1.

Table 2-1. Facilities Used to Develop Case Study Applications

LABORATORY	APPLICATIONS	SOFTWARE	HARDWARE	DATA SOURCE
Real Time Data System (RTDS) Lab Mission Operations Directorate Systems Division Communications and Data Systems Branch	Space Shuttle • Guidance, Navigation & Control (GNC) Intelligent Systems • Instrumentation and Communications Officer (INCO) Expert System Project (IESP) • KU Band Self Test Expert System • DATA COMM Expert System • Payload Deployment and Retrieval System (PDRS) Decision Support Sys.	G2 CLIPS C X Window System Masscomp graphics Comp Development Environment (CODE), Real Time Interactive Display Environment (RTIDE) UNIX	Workstations: DEC 3100 Masscomp (5600 and 6600)	Loral Instrumentation Advanced Decommutation System (ADS-100), RTDS custom data acquisition software
Artificial Intelligence (AI) Lab Information Systems Directorate Information Technology Division Software Technology Branch	Space Shuttle Onboard Navigation (ONAV) Expert System	C CLIPS X Windows System (early development using CURSES and Suntools) MC graphics in MPSR UNIX	PC Sun Workstation (development) Masscomp (delivery)	Telemetry from Real Time LAN Trajectory from General Data Request
Intelligent Systems Branch (ISB) Lab Engineering Directorate Automation and Robotics Division	Space Shuttle • Rendezvous Expert System (REX) • Payload Deployment and Retrieval System (PDRS) Decision Support System	LISP Flavors Joshua expert system shell CLIPS G2 UNIX	Symbolics 3650 Sun 4 workstation Masscomp (5600 & 6600) workstations	Data recorded in Shuttle Engineering Simulator and Shuttle Mission Simulator
Data Management System (DMS) Test Bed Engineering Directorate Automation and Robotics Division Intelligent Systems Branch (ISB)	Space Station Operations Management System (OMS) Prototypes	Initial ART and LISP Current ART/Ada Ada C X Window System OASIS	Initial Symbolics 3600 Current VAX workstation	Simulations resident on DMS Test Bed
Remotely Augumented Vehicle (RAV) Lab NASA Ames-Dryden NASA Integrated Test Facility	X-29 Remotely Augmented Vehicle Expert Systems (RAVES)	C VI DataViews UNIX	Masscomp workstation	RTDS data acquisition capability (see above)
STOL Maneuver Technology Demonstrator Edwards Air Force Base Air Force Flight Test Center	STOL Real-Time Interactive Monitoring Systems (RTIMES)	C CLIPS UNIX	Masscomp workstation	RTDS data acquisition capability (see above)

Included in this table are the hardware and software tools®™ used to build the intelligent systems.

2.2 Intelligent System and Functions

The following applications were evaluated in the case study:

- Real-Time Data Systems (RTDS) Applications
  Johnson Space Center
  - Space Shuttle Guidance, Navigation & Control (GNC) Intelligent Systems
    Monitor for orbiter sensors and Reaction Control System (RCS) jets and to detect Loss of Control of vehicle during ascent
  - Space Shuttle Instrumentation and Communications Officer (INCO) Expert System Project (IESP)
    Monitor of command and telemetry paths
  - Space Shuttle KU Band Self Test Expert System
    Monitor of checkout procedures for KU band radar
  - Space Shuttle DATA COMM Expert System
    Monitor telemetry recording by the onboard flight recorders and downlist of telemetry from these recorders
  - Space Shuttle Payload Deployment and Retrieval System (PDRS) Decision Support System (DESSY)
    Monitor of the subsystems for operation of the Space Shuttle Remote Manipulator System (RMS)
- X-29 Remotely Augmented Vehicle Expert Systems (RAVES)
  Ames Research Center's (ARC) Dryden Flight Research Facility
  Assist real-time, ground-based trajectory control of X-29 aircraft
- Military Aircraft Short Take Off and Landing (STOL) Real-Time Interactive Monitoring Systems (RTIMES)
  Edwards Air Force Base (EAFB)
  Monitor of thrust nozzle control during Short Take Off and Landing (STOL) of aircraft
- Space Shuttle Onboard Navigation (ONAV) expert system
  Johnson Space Center
  Monitor navigation state and sensors during entry

® - Ada is a registered trademark of United States Department of Defense
SUN and Suntools are registered trademarks of Sun Microsystems Inc.
VAX and DEC are registered trademarks of Digital Equipment Corporation

™ - ART is a trademark of Inference Corporation
Joshua is a trademark of Symbolics, Inc.
Symbolics is a trademark of Symbolics, Inc.
TAE Plus is a trademark of the National Aeronautics and Space Administration

- Space Shuttle Rendezvous Expert System (REX)
  Johnson Space Center
  Monitor procedure execution during Space Shuttle rendezvous and proximity operations

- Space Station Operations Management System (OMS) Prototypes
  Johnson Space Center
  Monitor for fault diagnosis, recovery planning, and execution of procedures for OMS

2.3 Human-Intelligent System Interaction Functions

Most of the applications in the case study assist operators in real-time monitoring tasks. Thus, agent activities primarily support detection and assessment of failures. The typical information monitored is state, status, commands, and configuration of the primary subsystems. Procedures are also frequently monitored (e.g., OMS Prototypes, REX, GNC Real-time Monitor, and KU Band Self Test Expert System).

A variety of types of fault information are monitored by the case study applications. The PDRS HCI design concepts provide access to information about faults indistinguishable using the available data (i.e., information about fault ambiguity). These design concepts also provide information for the integrated assessment of failures affecting more than one subsystem. Information that distinguishes faults from misconfigurations that occur when switching to redundant or alternate capability was identified during development of the IESP as important to fault management. Access to the history of component failures in previous missions is a planned enhancement to the IESP based on use of the original prototype. An assessment of the criticality of failures resulting in the "fail safe" recommendation is included in RTIMES.

In addition to monitoring, a number of other modes of operation were observed:

- Control
  A few applications include the ability to control the monitored process. Both the OMS Prototypes and REX have control modes, where procedures are automatically executed. RAVES assists ground-based aircraft control. It is planned that the DATA COMM Expert System will evolve from an intelligent assistant to a fully autonomous system that controls onboard data recording without human intervention. Such evolution will be gradual, for the risk of erroneous command entry by autonomous software results in tough reliability and safety requirements.

- Prediction
  Prediction capability is included in some cases. The manual mode of GNC Jet Control provides WHAT-IF analysis. It is used by operators to evaluate jet thruster reconfiguration after a loss of thrusting capability and to determine the potential for a failure to propagate. Both RTIMES and GNC Loss of Control explicitly identify the potential for a failure to occur.

- Review and Playback
  Most systems record some sort of information (i.e., logged information). Two classes of information are logged (1) data input to the intelligent system, and (2) information output by the intelligent system. Input data can be played back for review or used to re-execute the intelligent system. Output information can be played back for review.

- Training

  Since current training includes frequent practice of flight support during simulated missions, a logical extension of these support systems is their use as stand-alone training systems. Data from simulations or recorded during actual missions are replayed through the intelligent system as a means of providing off-line training to flight controllers (e.g., GNC Air Data System, ONAV). RAVES incorporates on-the-job training into the real-time support system by providing alternate display formats for novices and experts. Novices are trained using a format displaying items with full English descriptors. They can compare these descriptors to the commonly used acronyms arranged in the same layout on another display format.

Human-intelligent system collaboration and development of a shared view of the situation are supported by providing access to additional information about a displayed item, often in the form of pre-defined, explanatory text (e.g., KU Band Expert System, IESP, GNC Air Data System, DATA COMM Expert System). Direct access mechanisms between related information items on the display also supports collaboration. A planned enhancement to IESP is to extend pre-defined explanation to include pointers to related information. These pointers could be either reference listings or software links. Linking together related information was observed in other cases as a means of providing access to associated information. The GNC Air Data System provides hypermedia connections between a variety of types of information about this subsystem, including flight rules, system descriptions, and schematics. The PDRS HCI design concepts relate schematics to associated tables of design information and associate fault ambiguities with procedures that can resolve the ambiguities.

The ability to intervene in intelligent system processing is generally limited. The PDRS HCI design concepts provides the greatest variety of operator intervention capabilities seen in the case study. One type of intervention is manipulating the information input to the intelligent system. The PDRS HCI design concepts provides the operator with the capability to disable parameters and to input information not available on the downlist (e.g., information on the voice loop) or internal to the intelligent system. Both RTIMES and the PDRS HCI design concepts allow filtering of telemetry parameters to assist in managing noisy data that can generate false alarms and misdirect the intelligent system. A second, more drastic type of intervention is restart of the intelligent system. Although a number of applications allow intelligent system restarts, none but the PDRS HCI design concepts include restart from checkpoint (i.e., saving internal state of intelligent system for use when re-executing the intelligent system from some time in the past).

Real-time performance problems can occur when applications with complex HCI are implemented using advanced software tools. Both GNC Air Data System and KU Band Self Test Expert System encountered such problems. The PDRS HCI design concepts provides information (e.g., Central Processing Unit (CPU) and memory usage) for monitoring intelligent system performance. Both the PDRS HCI design concepts and REX provide operational capability useful in improving performance (e.g., ability to disable an incorrect or unnecessary rule base).

2.4 Supporting User Interface Capabilities

A wide variety of user interface capabilities were observed in the case study. Most applications conform, however, to the following common characteristics:

- Direct manipulation of display items
- Message lists for display of information from intelligent system
- Scrolling as a review mechanism
- Graphical schematics displaying component status or availability
- Color for coding, especially status (e.g., good, cautionary, or bad); redundant coding often used with color (e.g., status stated in text and indicated by color)

Many of the user interfaces use graphical items or display formats from the current flight support environment. The availability of hardware platforms supporting graphics allows emulation of current paper-based formats or off-line capabilities in an on-line, electronic form modifiable in real-time by raw telemetry or operator input. Examples include:

- RTIMES strip charts
- GNC jet control diagram
- PDRS Position Monitor's three dimensional projection of vehicle, arm, and payload
- ONAV Manual Select Keyboard (MSK) and Digital Display Device (DDD)
- Portions of MSK for GNC Air Data System
- DATA COMM data tabs format and recorder management table
- REX activity timelines

The behavior of the user interface is often dependent upon the mode of operation of the intelligent system (see GNC Real-time Monitor, GNC Jet Control, KU Band Self Test Expert System, and the PDRS HCI design concept). The available information and operational capabilities vary with the mode of operation. For example, mouse-selection of the jet thruster icons on the diagram used for GNC Jet Control resulted in display of different information and access to different control options, depending upon the currently active mode. The format of the display was also observed to vary with mode of operation. For example, the background of the workspace for the PDRS HCI design concepts varies with the selected mode.

The behavior of the user interface is also controlled by the operator in some applications. The PDRS Temperature Monitor and OPS Monitor provide alternate ways to interact with the user interface (i.e., either mouse inputs or function keys). REX allows explicit operator control of display configuration, including the ability to save a display configuration and switch to another display configuration. Many applications partitioned the workspace into fixed and dynamic regions (e.g., PDRS HCI design concepts, IESP, KU band, GNC Air Data Systems, REX), where the contents of the dynamic regions was selected by the operator.

The source, quality, and availability of an information item can be useful in interpreting the item (see section 3.2.3 in Volume 1, Malin et al., 1991). A variety of techniques are used to display these information attributes or qualifiers. Methods used to explicitly identify the source of information include displaying a distinctive color to identify a source (e.g., GNC Real-time Monitor) and listing the source name in messages (e.g., REX and PDRS HCI design concepts). Redundant sources can be simultaneously displayed for comparison (e.g., GNC Real-time Monitor). Static data due to Loss Of Signal (LOS) are indicated by placing an "s" next to the displayed parameter for GNC Real-time Monitor and by displaying an unique color border to indicate static data (i.e., orange) for the PDRS HCI design concepts. A planned enhancement of the ONAV system is to indicate the data quality of a parameter on plots of that parameter by only plotting data with good data quality.

A number of techniques that assist the operator in identifying relationships between displayed items were observed in the case study. Items not collocated on the screen are related by using the same background coding (e.g., color, pattern). For the PDRS HCI design concepts, a background pattern is used to associate a popup window with the manipulation device used to call up the window. Maps or schematic overviews are used by the PDRS HCI design concepts to identify the displayed portion of a schematic or workspace in the context of the total schematic or workspace available for display. In REX, operators can create relationships by annotating display items. Notes can be associated with an activity or at a specific time on the timeline form. Alternately, reminders, indicated by a yellow sticky icon, can be attached to a specific region of the timeline display space.

Methods for quick assessment of the current situation were observed in a number of cases. The GNC Loss of Control uses coding of border color to quickly orient operator about the potential for loss of vehicle control. The DATA COMM Expert System provides a window for quick look assessment by plotting key parameters (e.g., percentage of tape recorded during AOS, percentage of tape recorded during LOS, head temperature) that constitute metrics for the health of the recording process. The Event Timeline in the PDRS HCI design concepts uses color coding of subsystem status to summarize integrated status (i.e., regions of red or yellow quickly identify subsystems with problems).

Methods to indicate the current position in the flight profile with the relevant flight rules (i.e., document that defines and specifies proper response to nominal and off-nominal conditions and behavior) are provided by both the GNC Air Data Systems and ONAV. Both systems explicitly represent and display flight rules. ONAV encodes flight rules as thresholds on plots of parameters that are monitored for compliance with the flight rule conditions. Both display key flight profile parameters (i.e., the altitude of the vehicle) for the entry phase. GNC Air Data System displays altitude on a meter while ONAV includes the current altitude in each displayed message.

Many systems provide mechanisms to minimize interruption of the operator by incoming information. Message buffering until the operator has time to view messages occurs in the GNC Real-time Monitor, KU Band Self Test Expert System, and DATA COMM Expert System. The DATA COMM Expert System alters the appearance of a message icon to indicate that information is waiting. Message priorities are used by IESP as a means of indicating the importance of a message to the operator. A possible extension of this message priority approach is to infer the importance of message and only display the important messages. REX provides optional voice enunciation of thrust commands to allow the crew to enter command without looking at screen.

The simultaneous display of predicted behavior and actual behavior was observed in the KU Band Self Test Expert System. A planned extension to this approach is to also display the behavior history. The display sequence of "previous, current, predicted" would clearly identify the position within a complex, sequential procedure such as the self test.

The display of flight-critical information independent of the intelligent system was frequently observed. Such partitioning of the display prevents total loss of flight support capability should the intelligent system fail and allows the operator to take over tasks typically assigned to the intelligent system. For the Space Shuttle applications, this was achieved by operating the intelligent system side-by-side with the existing displays. For RAVES, the integrated set of screen formats included displays that were active even when the intelligent system was disabled.

2.5 Design Process

Many of the prototypes evaluated by the JSC study team were built for Space Shuttle flight support. This results in a description of the design process that is heavily influenced by the Space Shuttle applications. Also, Space Shuttle prototypes frequently have the goal of operational capability. The Space Station applications are recognized as prototypes for the purposes of requirements development and do not focus on issues of certification and configuration management that are necessary for operational systems.

A predominant number of applications are RTDS prototypes. Most of these systems followed a user-driven design process. Also, when implementers were not flight controllers, they were often from a small group of software developers who had worked on a number of RTDS systems. This promotes reuse of software and commonality of design methodology and style.

A number of common traits were observed in the course of the case study and thus represent attributes of the typical design process for intelligent systems for aerospace flight support. Most of the systems were built to support either the crew or the flight controllers. Thus, in many of these cases, the users are the domain experts. Few formal, written requirements were generated, with the exception of REX and ONAV (where requirements were generated after initial prototyping). Most requirements were identified in working meetings between domain experts, users, and software developers. For the cases of user-generated software, one person often fulfilled all roles. In most Space Shuttle cases, early, active user involvement in the design and development was observed. For Space Station prototypes, the domain experts (typically design engineers) were the early participants, since the prototypes are targeted for requirements development and not operational use. An iterative development process was observed, consisting of demonstration and testing, evaluation of the results of tests, and modification of the prototype. Frequently, testing included user review. When the user group contained a number of members (such as a flight control position with multiple trained personnel), a small subset of the user group was continuously involved and the remainder of the user group were only involved at testing and review. When integrating with existing software (as for the Space Shuttle prototypes), new HCI designs typically evolved from the current displays. This evolution was gradual and occurred as small, frequent upgrades. When possible, testing was performed in an environment as similar to the operational environment as possible (e.g., REX is integrated with Space Shuttle Engineering Simulation (SES), RTDS applications are used in parallel operations during integrated training simulations in the MCC). Side-by-side operation of the existing support displays with the new systems and displays was frequently used, either by testing in the operational environment (e.g., RTDS applications) or by emulation of the existing displays (e.g., ONAV). There was no standard certification process, although most involved user evaluation of system accuracy in a variety of error situations (such as support of integrated training simulations). No system in the case study was sufficiently mature to have addressed issues of integration with the existing software maintenance and configuration management systems.

Distinguishing characteristics were also observed. The skill level of the implementers varied from experienced software developers to untrained users. Performance was frequently a problem with software developed by untrained users. This resulted from the high overhead typically associated with advanced HCI tools (e.g., the X Windows System™) and expert system shells (e.g., G2®). Achieving real-time performance for complex HCI using these tools often required a level of development experience that the untrained user did not possess. Although user involvement was common, the role of the user in the design process varied

™ - the X Windows System is a trademark of MIT.

® - G2 is a registered trademark of Gensym Corporation widely. In some cases the user served as domain expert, in others as software designer and implementer. The cases differed in the scope of the application as well. Some prototypes were designed to support a very specific monitoring task (e.g., GNC Loss of Control Expert System detects loss of control of Space Shuttle during ascent, based on monitoring of pitch and yaw angles) while others were designed to support the flight controller during multiple flight support tasks (e.g., the ONAV Expert System assists controllers in assessing the quality of measurements from multiple navigation sensors, the compliance of the navigation state with flight rule limitations, the quality of the navigation state vector computed onboard the vehicle with respect to the redundant ground-computed state vector).

The remainder of this section will discuss some of the unique design approaches observed during the case study.

RAVES was designed to allow users to generate custom display designs for the operational application. This was accomplished by providing users with a display design tool (i.e., RAVES chose DataViews®) that is easy to learn and that can generate a description of the design usable for implementing the display. This description consists of a picture of the visual screen design and a list of the parameters to be displayed, called a data source list (CSC, 1990). System implementers then modify the operational system to include the new display. Such a process is one way of allowing users to design displays while ensuring that the resulting software will comply with accepted software standards for development and configuration management.

As described in section 6 of Volume 1 (Malin et al., 1991), evaluation using operational scenarios is necessary to identify the full range of necessary information and agent activities. INCO used operational scenarios to evaluate the design early in the design process. The RTDS philosophy of early operational testing in parallel with the existing system is a form of evaluation using operational scenarios. For the PDRS Decision Support System, interactive displays based on the HCI design concepts were implemented using a software prototyping tool as a vehicle for users to perform hands-on evaluation of the design.

The storyboard prototyping technique used to develop the PDRS HCI design concepts was an effective means of identifying some HCI requirements. ONAV provides an example of an interesting approach to requirements documentation. Formal requirements documents were developed after the initial prototype was built to serve as the repository for knowledge gained during prototyping. This approach represents the translation of a prototype into requirements, although this prototype also evolved to an operational system. In both cases, written requirements were derived from the prototype manually without assistance from explicit design methods or tools.

The translation of the PDRS design concepts from a storyboard format to an electronic prototype required that a number of compromises be made. The user interface prototyping tool imposed a variety of design constraints based on its capability. When the prototyping tool is not the same as the delivery tool, these constraints may be unconsciously and unnecessarily incorporated into the delivery system, unless such constraints are documented.

Human factors expertise was not typically available during the development of most of the intelligent systems in the case study. Development of the PDRS HCI design concepts, however, included review by human factors personnel (i.e., J. Malin and M. Czerwinski, see the appendix). To partially compensate for this lack of expertise, HCI design guidance for intelligent system designers is needed. A format was developed for the delivery of design

® - Data Views is a registered trademark of VI Corp.

guidance examples to the PDRS system developers. This format was also tested in providing design guidance to the OMS Prototypes.

Simulation was frequently used as a means of generating data for testing applications. Most developers use existing simulations (e.g., SES, Space Shuttle Mission Simulator (SMS)). The KU Band Self Test Expert System developed a stand-alone simulation in G2, however, for early testing. This capability allows selective failing of portions of the checkout procedure for use during testing of the intelligent system and would be useful for building cases to train operators on operational use of the system. Simulation may also be used to implement the operational scenarios (i.e., expected sequences of activities during operations) discussed previously for use during intelligent system design.

Performance problems were observed in some of the RTDS applications. The GNC Jet Control system was tuned by a software engineer to avoid these performance problems. This development approach differed from the typical development model of rapid prototyping with iterative upgrades.

A number of certification techniques were observed, including (1) iterative demonstration and review by users, (2) use in parallel with existing displays, and (3) use of simulation or recorded data for off-line testing. The lack of formal requirements has made certification difficult, for typically software is certified by formal tests to verify that requirements have been met. ONAV has a detailed certification plan, reflecting its status as one of the first intelligent systems to address integration with the operational configuration management and data delivery system. Since RTDS uses its own stand-alone data acquisition system, it has bypassed the need to integrate with the operational configuration management and data delivery system. The plan to certify ONAV consists of:

- Certification of the data acquisition software (i.e., software written in C to strip data from the LAN, load it into shared memory, and preprocess it prior to intelligent system processing)

- Certification of the rule base, using a matrix of possible failures; the system must function correctly on at least two test cases for each specified failure

- Certification of the usability of the user interface in the operational environment; this must wait until the tools necessary for delivery of the final user interface (i.e., the X Windows System) are available under the configuration manager for operational software

- Final certification of the full system by providing support successfully during a training session of integrated entry simulations (i.e., approximately 12 different simulation runs of the landing portion of a Space Shuttle mission).

2.6 Summary of Issues

Design issues requiring further investigation were identified during the case study. In brief, these issues are:

- Support for multi-tasking dynamic task assignment
  Applications were frequently designed to support the operator during a variety of fault managements tasks. Such broadly scoped intelligent systems should assist the operator in selecting tasks and in altering activity sequences as situation changes. Operational modes were commonly provided as a means of altering activity sequence. How system capability is altered at a mode change or even when a mode change has occurred is not always clear, however. Many systems provide some assistance in handling interruptions of the operator when new information becomes available (e.g., message priorities, buffered messages), but additional work is needed, especially in the areas of suspending and resuming activities during multi-tasking. Designing the intelligent system to effectively coordinate with the operator during joint tasking is also an issue that was not addressed by applications in the case study.

- Collaboration between the operator and the intelligent system
  Communication with the intelligent systems of the case study is limited to explanation consisting of pre-defined text descriptions or pointers to related information. Such communication is effectively one way. Two-way exchange of information between the operator and the intelligent system is required to develop a shared view of the world. Some support for developing this shared view is provided using review of information history (especially for the monitored process). There is a need to also identify behavioral trends in that information over time.

- Capability to intervene into and control the intelligent system
  The ability to intervene in intelligent system processing typically consists of restarting the intelligent system. In some applications, input data are filtered to minimize the need for such intervention (i.e., reduce impact of noisy or erroneous data on the intelligent system). There is little support for redirection of the intelligent system or real-time alteration of task responsibilities.

- Interpretation of displayed information
  Many of the case study applications assist the operator in interpreting information by qualifying displayed information items with either source, quality, or timing (i.e., availability or ordering of information) assessments. In some cases, a related event is provided as context for interpreting information (e.g., altitude was associated with fault messages in ONAV). There is a general recognition that such additional interpretive context is useful, but information presentation techniques that do not contribute to information overload of the operator are needed.

- Support of fault management tasks other than data monitoring
  Data monitoring was the most common fault management task performed by the intelligent systems surveyed. Other important fault management tasks include monitoring and execution of activity sequences (i.e., procedures) and replanning activities (including impact assessment) when a failure occurs. Research is needed to identify the information required for these tasks and effective methods of presenting this information.

- Use of intelligent systems designed for real-time operations for off-line training
  Many applications were mentioned as potential systems for off-line training capability as well as real-time operations. Yet most systems were designed for real-time use. Investigation into building systems that can effectively support real-time operations while addressing training requirements is needed.

- Compensation for performance problems
  Some of the applications had performance problems. Approaches to solving these problems included fine-tuning the implementation and using the system only for off-line support (i.e., in situations not requiring real-time response). Techniques for identifying and correcting potential intelligent system performance problems during design are needed. Additionally, designs that permit real-time adjustment of performance are needed (e.g., REX allows selective disabling of portions of the system in real time). This becomes more important with the trend toward user-written software. Because users are often not trained in software development, this development approach can lead to performance problems in the implemented system.

- Characteristics of the design process
  The intelligent system design process in the cases studied is characterized by early user involvement, iterative prototyping with gradual development of the design, and testing in operational environments. A development methodology and tools are needed to support such a design process and to develop an explicit specification of user requirements.

- Use of available user interface design guidelines
  Few developers used existing user interface guidelines documents to design the user interface. The reasons cited include the difficulty of using these documents and the limited development time available for using them. Electronic availability of guidelines and examples, especially if integrated with a development tool, is considered as a requirement for effective use of guidelines. A development methodology and tools are needed that makes use of design guidance easier and integral to the development process.

- Common user interface designs
  Most user interfaces are direct manipulation interfaces. They are characterized by multiple windows, frequently where one window overlays another. Message lists and schematics are commonly used in the cases studied. Color coding is also heavily used. Based on these user interface trends, further research on window management, message lists, schematics, and color coding would be particularly useful.

2.7 Case Data Sources

CSC (April, 1990), *Remotely Augmented Vehicle Expert System User's Guide*, TM-4000-04-01, Revision 1, prepared for NASA by Computer Sciences Corporation.

JSC (October, 1983), *PDRS Console Handbook, Volume I*, JSC-17525, Johnson Space Center, Houston, TX: NASA.

Malin, J. T., D. L.Schreckenghost, D. D. Woods, S. S. Potter, L. Johannesen, M. Holloway, and K. D. Forbus (September, 1991), *Making Intelligent Systems Team Players: Case Studies and Design Issues, Volume 1. Human-Computer Interaction Design*, NASA Technical Memo, Houston, TX: NASA - Johnson Space Center.

Rasmussen, A.N., J.F. Muratore, T.A. Heindel (August, 1990), "The INCO Expert System Project: CLIPS in Shuttle Mission Control", *Proceedings of First CLIPS Conference*, NCP 10049, Volume 1, Johnson Space Center, Houston, TX: NASA.

Section 3
Guidance, Navigation, and Control (GNC) Applications

3.1 System Description

Four applications that use RTDS have been developed for the GNC flight controllers over the last year. These systems are:

- GNC Real-time Monitor
- GNC Jet Control
- Air Data System
- Loss of Control (LOC)

These systems support GNC flight controllers during various phases of a Space Shuttle mission. Integration of these applications remains an issue.

All systems were demonstrated on a DEC® 3100 workstation located in the RTDS development laboratory. This laboratory contains workstations connected to the RTDS data acquisition network, so prototypes can be tested in real-time operations during training simulations or missions. A VAX® 2000 workstation is used in the Mission Operations Center (MOC). To improve performance during flight support for systems developed using G2®, these systems are executed on a VAX 3100 workstation and displays are networked to the VAX 2000.

The GNC applications are built using a variety of software tools. The GNC Real-time Monitor is implemented in C with an user interface built on the X Windows System™. The GNC Jet Control application is developed in G2. This application consists of 80-90 rules for the Digital Auto Pilot (DAP) sub-mode evaluations and 25 rules for display. Both the forward and backward chaining capability of G2 were used. The Air Data system is developed in G2 and consists of several hundred rules. The Loss of Control application was developed as a rule-based system in G2. It currently contains 10-15 rules.

3.2 Intelligent System and Functions

GNC Real-time Monitor

This application was built for monitoring critical parameters affecting the GNC systems. One process in the GNC Real-time Monitor assists the operator in determining the status of a startracker check out procedure (i.e., *self-test*). A startracker is a sensor onboard the Space Shuttle that senses visible light. It is used to inertially align the Inertial Measurement Units (IMUs) to star positions and to track a rendezvous target using reflected light. A second process compares the acceleration measurements from all IMUs to detect anomalous measurements. Three IMUs are available onboard the Space Shuttle for use in both translational and rotational navigation. The third process monitors Orbital Maneuvering System (OMS) gimbal angles for anomalies during OMS burns that occur onorbit. The fourth system assists the operator in monitoring elapsed time during a vehicle maneuver.

® - DEC is a registered trademark of Digital Equipment Corp.
® - VAX is a registered trademark of Digital Equipment Corp.
® - G2 is a registered trademark of Gensym Corp.
™ - The X Window System is a trademark of MIT

GNC Jet Control

The GNC Jet Control application is a rule-based system that assists the GNC flight controllers in monitoring the Space Shuttle primary Reaction Control System (RCS) used during maneuvers and burns. The status and configuration of the RCS jets and the Muxer-DeMuxers (MDMs) connecting them is assessed. This application is also used to assess valid DAP sub-modes for a given jet configuration.

The GNC Jet Control application can also be used to perform what-if investigation concerning the RCS system. Jets and MDMs can be selectively failed (within the application only) and the impact of these failures assessed.

Air Data System

The Air Data System is a rule-based system developed to monitor the Air Data Probe sensors used to collect barometric data for navigation during Space Shuttle entry. The state and status of component of the air data probes are assessed and the flight profile is monitored to identify when to take air data. Additional background information about each component is also accessible to the operator.

Loss of Control

The Loss of Control application is a rule-based system that assesses the potential for loss of control of the vehicle during ascent. This information is critical in making mission abort assessments. Loss of control of the vehicle when over a populous area can result in a catastrophic abort (i.e., destroy the vehicle).

3.3 Human-Intelligent System Interaction Functions

The GNC applications are used primarily for monitoring, with some support for planning also. None of the GNC applications provide the capability to intervene with or control the monitored process from the intelligent system.

GNC Real-time Monitor

The GNC Real-time Monitor consists of four processes:

- Startracker Self-test
  Monitors telemetry relevant to the startracker self-tests to determine status of the self-test. It also allows playback of the last self-test for closer evaluation of each step in the test. Operator controls include an optional Y or -Z startracker analysis window and stepwise movement through data from a self-test.

- IMU Acceleration
  Compares redundant acceleration measurements in IMU reference coordinates from the three available IMUs. The current display capability is limited to viewing one IMU at a time. Reference frame is specific to a vehicle and therefore will change from mission to mission.

- OMS Thrust Vector Control (TVC)
  Monitors telemetry relevant to the OMS for movements of OMS gimbal angles that affect post-MECO (Main Engine Cut Off) OMS burns. Operator controls include an optional display analysis window.

- Maneuver

  Computes and displays time to burn or maneuver at the current rotation rate. This process replaces an existing manual procedure that was done by visually monitoring raw data at a 1 second data rate. The operator can change elapsed time by one second increments.

The information from these processes are displayed in two forms, as messages or as tabulated data.

There are two modes associated with the both the Startracker Self-test and the OMS TVC: (1) real-time monitoring and (2) playback of data. To allow playback, the operator must "turn off interruptions", which buffers incoming. real-time data. During playback, the operator can step through stored information for a closer examination.

GNC Jet Control

The GNC Jet Control application has two distinct uses:

1) Monitoring

   To assess the current primary jet configuration for the Space Shuttle propulsion system and the valid DAP sub-modes for that configuration. It also detects jet and MDM failures and determines how those failures impact DAP sub-modes.

2) Planning

   To allow what-if investigations of the effect of changes of jet or MDM configuration on DAP sub-modes
   - Checking planned nominal configurations
   - Reconfiguring after jet or MDM failures
   - Evaluating the impact of potential failures (i.e., next failure analysis)

There are two modes of operation that correspond to the two uses listed above: automatic and manual. In automatic mode, real-time telemetry data are displayed for monitoring, actual jet failures are presented on a diagram showing jet configuration, and actual MDM failures are displayed using status panels. In manual mode, the current configuration is frozen to allow the user to investigate the effects of reconfiguration while monitoring continues in the background. This investigation includes manually "failing" jets or MDMs to predict the impact of such failures.

During manual mode, real-time monitoring continues in the background. Anomalies are reported to the operator in a message panel, but are not reflected in the diagrams.

This tool is especially effective in evaluating DAP sub-modes during multiple jet failures, since multiple failures increase the complexity of this task significantly. Currently, a matrix of configurations is used for single failures and manual procedures are used to evaluate DAP sub-modes for multiple jet failures.

Air Data System

The Air Data System assesses state and status of all components of the Air Data Probe sensors. State and status are displayed on a schematic of the air data probe sensors. The raw telemetry used to determine state and status are accessible via a table. Parameters critical in determining when to take air data (i.e., speed, altitude, and orientation with respect to flight path) are displayed on sliding scales with important event thresholds clearly marked. Messages are logged and critical messages are "popped up" on a separate window to attract operator attention.

Support for collaboration includes allowing the operator to request additional information about each component of the schematic, including pre-defined text explanations, related flight rules, a more detailed schematic, and either tabulated or plotted data.

The system does not execute sufficiently fast to support real-time operations, so it is currently not used for flight support. It does have potential as a tool for training novice flight controllers on the details of the air data probes, since it provides access to subsystem descriptions, related flight rules, and detailed schematics. At the time of the interview, however, the system was not being used in this capacity.

Loss of Control

The Loss of Control application monitors two telemetry values, the vehicle pitch and yaw angle rates, continuously during the ascent phase of a mission to assess the status of vehicle control. If these parameters change at a rate exceeding a pre-defined threshold for longer than is deemed safe, an alarm is raised. Specifically, the potential for loss of control occurs when the rate of change in either the pitch or yaw angle exceeds 3 degrees per seconds. If the rate of change exceeds 5 degrees per second for over 5 consecutive seconds, loss of vehicle control has occurred.

The status of vehicle control is presented via messages, with supporting data about angle rates and elapsed time since angle rate limit was exceeded. Color is also used to indicate status of vehicle control. The operator can optionally access a history of angle rate values in the form of a data plot.

3.4 Supporting User Interface Capabilities

GNC Real-time Monitor

The application has five overlapping, translatable windows corresponding to one window for each of the processes in the GNC Real-time Monitor, plus an additional window for displaying messages from all of these processes (see figure 3-1). Each of the four processes generates timetagged messages that are displayed chronologically in the message list. Currently the message list is not scrollable, but scrolling is a planned enhancement.

The information displayed within each process window is formatted similarly to current displays, with the addition of new data and color. Color text is used to indicate the source of data (i.e., raw data is yellow, titles are brown). A possible future enhancement is the use of color to indicate static information. Currently static information is indicated by appending an "s" to the data value (i.e., an emulation of the existing display format).

The quality indicator shown on the specification bar at the top is a numeric value provided by the RTDS data acquisition software. It represents an estimate of data reliability and quality and ranges from 0 (worst quality) to 100 (best quality). Flight controllers monitor this parameter to determine if observed data problems could be due to the data acquisition process.

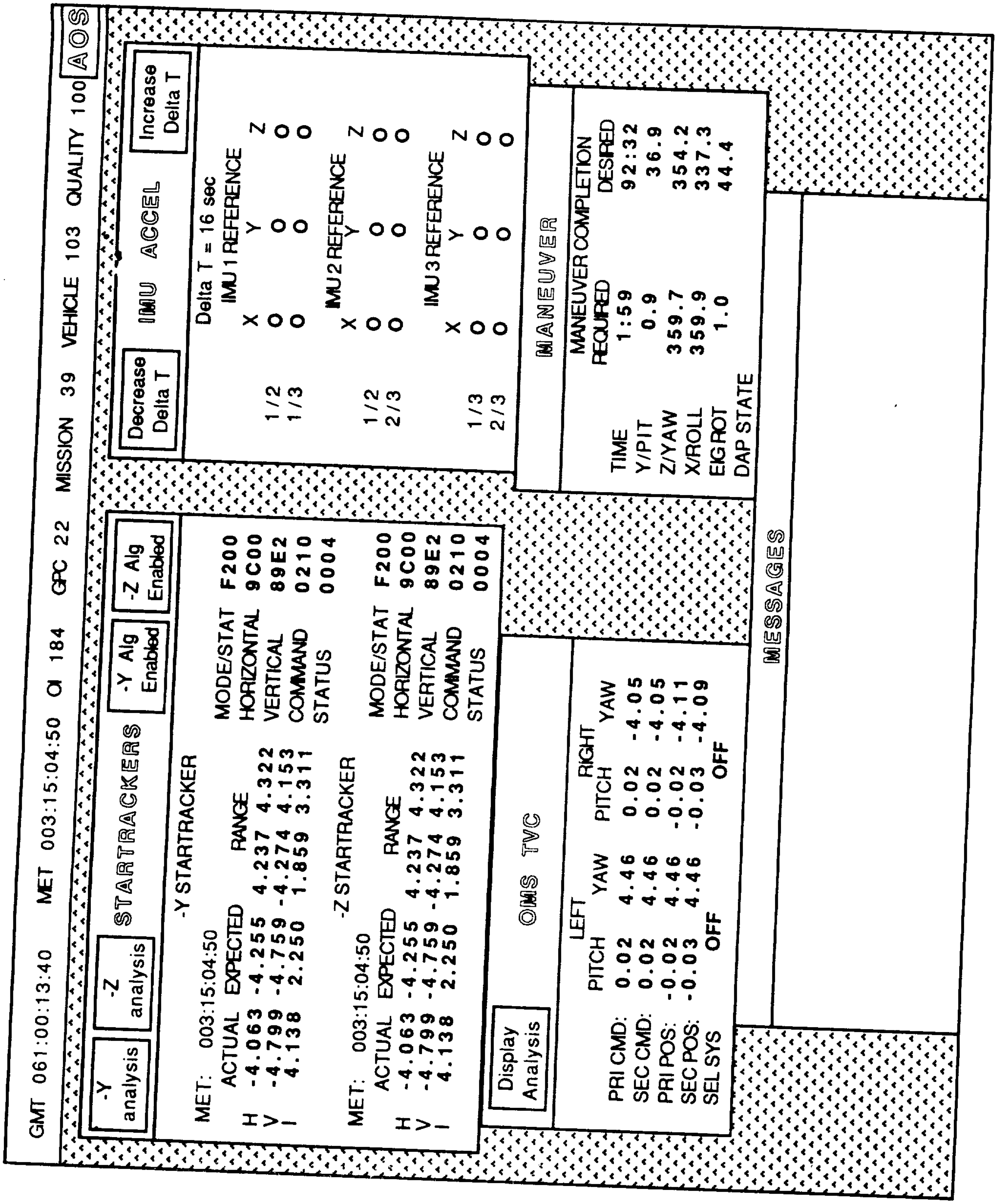


Figure 3-1. GNC Real-time Monitor

Three additional popup message windows can be accessed from the title bars of the process windows, specifically -Y or -Z Startracker analysis window from Startracker Self-Test and OMS TVC display analysis window from OMS TVC. These windows provide detailed messages concerning the information in the window from which the popup was enabled. Operator capabilities available from these popup windows includes:

- Enable or disable interruptions (i.e., display or buffer new messages)
- Hardcopy display
- Scroll messages forward or backward
- Close window

All popup windows are translatable.

During a startracker self-test, 10 seconds of data are stored in a file accessed by vehicle identifier for later evaluation. This analysis is conducted from the startracker analysis windows by disabling interruptions, which automatically loads the last data set from a star tracker self-test and displays the associated values in the main window (see figure 3-2). The user can step chronologically through the data set and the appropriate messages and data values are be displayed dynamically. This allows closer evaluation of the results of the startracker self-test, which are downlisted as HEX values. This was prompted by a frequent need to playback data from these self tests during actual mission support. Notice that playback analysis can only occur if interruptions are turned off (i.e., current data are buffered). If not turned off, the buttons to step through playback are not displayed.

The OMS TVC analysis window displays detailed messages from monitoring of the ascent OMS gimbal angles. It has the scrolling, interruption disable, and hardcopy capabilities of the startracker analysis windows.

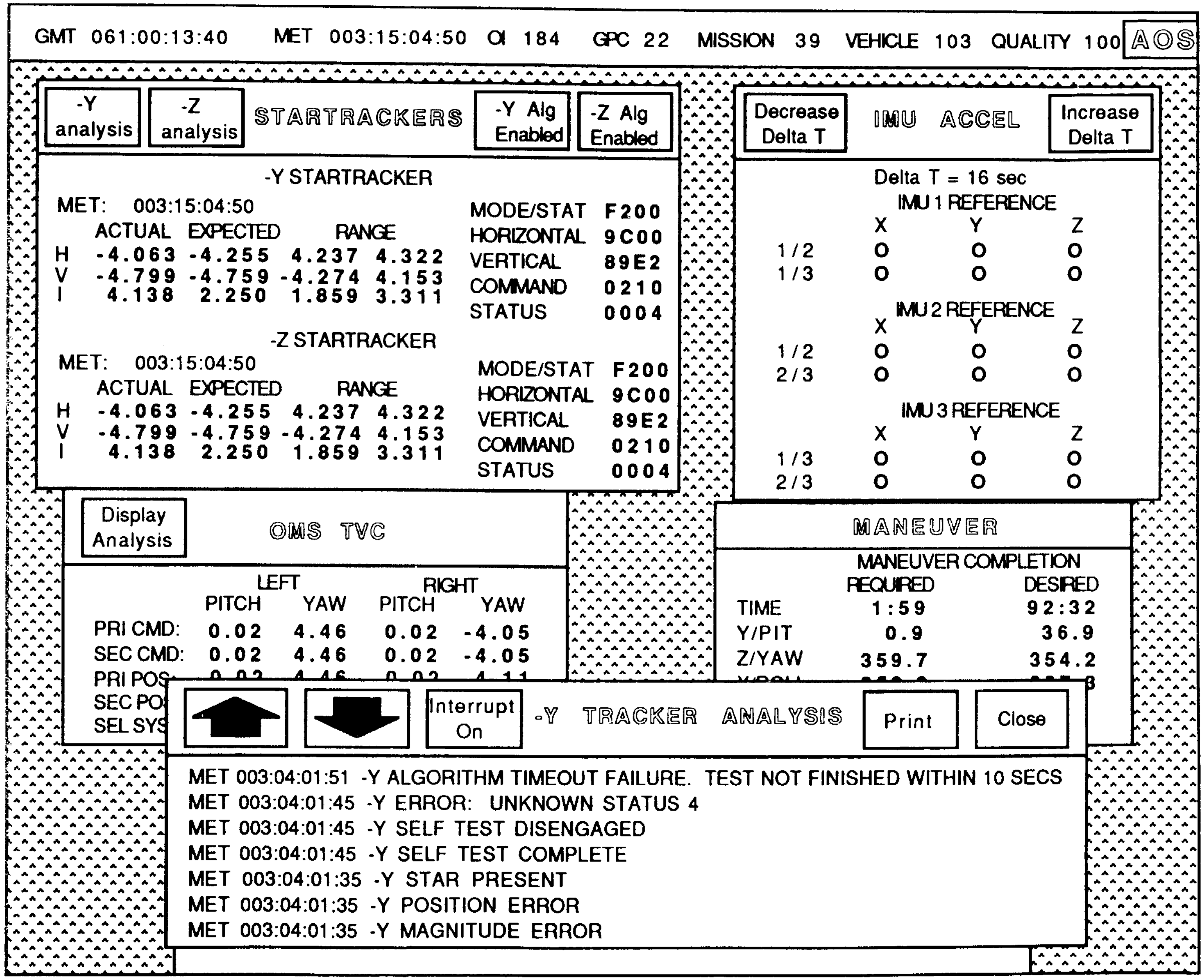


Figure 3-2. GNC Real-time Monitor with Startracker Analysis Active

GNC Jet Control

The basic display consists of a graphic of all Space Shuttle jets and two windows containing valid DAP sub-modes, one for sub-modes during vehicle rotation and the other for sub-modes during vehicle translation (see figure 3-3). There are 18 possible sub-modes each for both rotation and translation. The graphic consists of jets represented as arrows pointing in a fixed direction on a coordinate system appropriate to burns and maneuvers (i.e., Space Shuttle body coordinates). Individual jets are clustered by location into groups. The status of a jet is indicated by the color of the arrow representing it on the graphic:

- Green - jet available
- Yellow - single jet failed
- Red - entire group of jets failed

Eight MDMs (specifically the Flight Forward (FF) 1-4, and Flight Aft (FA) 1-4) connect these jet groups to allow communication with the General Purpose Computers (GPC). The status of these MDMs is also indicated in a status matrix in the lower left of the screen (i.e., red block for bad MDM and green block for good MDM).

This graphic duplicates a drawing from current flight support manuals. Operators currently place a paper copy of this graphic in a plastic cover, then mark failures with colored grease pens to assist in the assessment of reconfiguration options. Operations in the manual mode of this application emulate these procedures by allowing the operator to mark jet failures on the graphic.

In the manual mode, the user can use the mouse to fail or "un-fail" the jets or the MDMs for checking variations on the current configuration with respect to the valid DAP sub-modes. Mouse selection of an item calls up a menu of possible status values. When the current status is changed, an assessment of the sub-modes for that configuration is generated in the two sub-mode windows. The possible assessments are:

- LOC
  Loss Of Control of vehicle
- Sloppy
  Wide variations in vehicle motion between deadbands (i.e., the pre-defined error tolerance in vehicle orientation with respect to the commanded position)
- High RCS
  High fuel consumption
- Good
  Meets control, fuel, and deadbanding constraints

Monitoring of the RCS and MDMs continues in the background and the operator is alerted of any changes in configuration by a message in the lower portion of the screen. No changes to the status indicated on graphic occur, however, until automatic mode is invoked.

Selection of an arrowhead in automatic mode calls up a table of parameters relevant to the associated jet. Parameter values include (1) the downlist parameter identifier, or Measurement Stimulus Identification (MSID), (2) the most current telemetry value, (3) the communication status (i.e., status of MDM determined by expert system), (4) the power status (actually a place holder for later enhancement), (5) the status of jet determined by expert system, and (6) the status of the MDM associated with the jet provided by the operator.

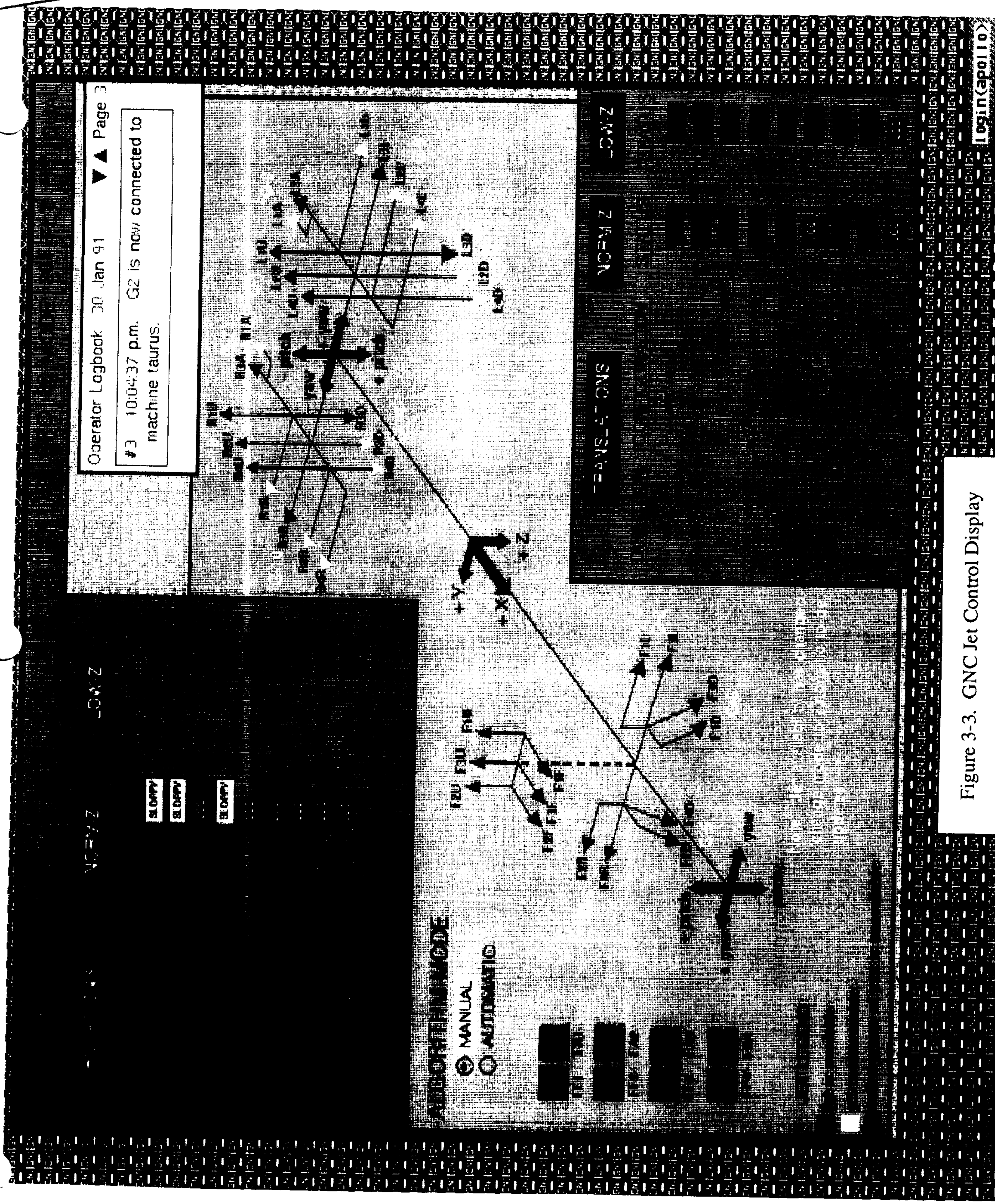


Figure 3-3. GNC Jet Control Display

ORIGINAL PAGE
COLOR PHOTOGRAPH

Air Data System

The user interface consists of a hierarchical series of windows, with increasing detail in information with increasing depth into the hierarchy. At the top level, five windows are provided (see figure 3-4):

1) Schematic of the air data probes

   The status of components of this system is determined by the rule-based system and indicated using color and status words displayed on subsystem components of the schematic. Multiple lines connect components and color distinguishes the type of connection. A legend describing the meaning of each color is included. Icons visually representing the state of the probes (i.e., stowed or deployed) and the state of related switches are also provided.

2) Emulation of current display

   Raw telemetry data are currently displayed in a tabular format on a Manual Select Keyboard (MSK)

3) Summary of flight profile

   Three key parameters indicate the vehicle's current location, speed, and orientation with respect to flight path:

   - Altitude
   - MACH number
   - Alpha (i.e., angle of vehicle with respect to flight path)

   Data for each parameter is displayed using a pointer on a sliding scale. Additionally, important thresholds are marked in red (e.g., expected time of air data probe deploy). All three scales are aligned horizontally, so relationships between parameters can be easily observed.

4) Operator log book

   Window that contains status messages sorted in chronological order. Instead of scrolling, fixed sized message windows are layered on top of each other as new messages arrive. A message consists of message number, timetag, and content of message.

5) Message board

   Window that pops up to alert the operator when critical messages are received.

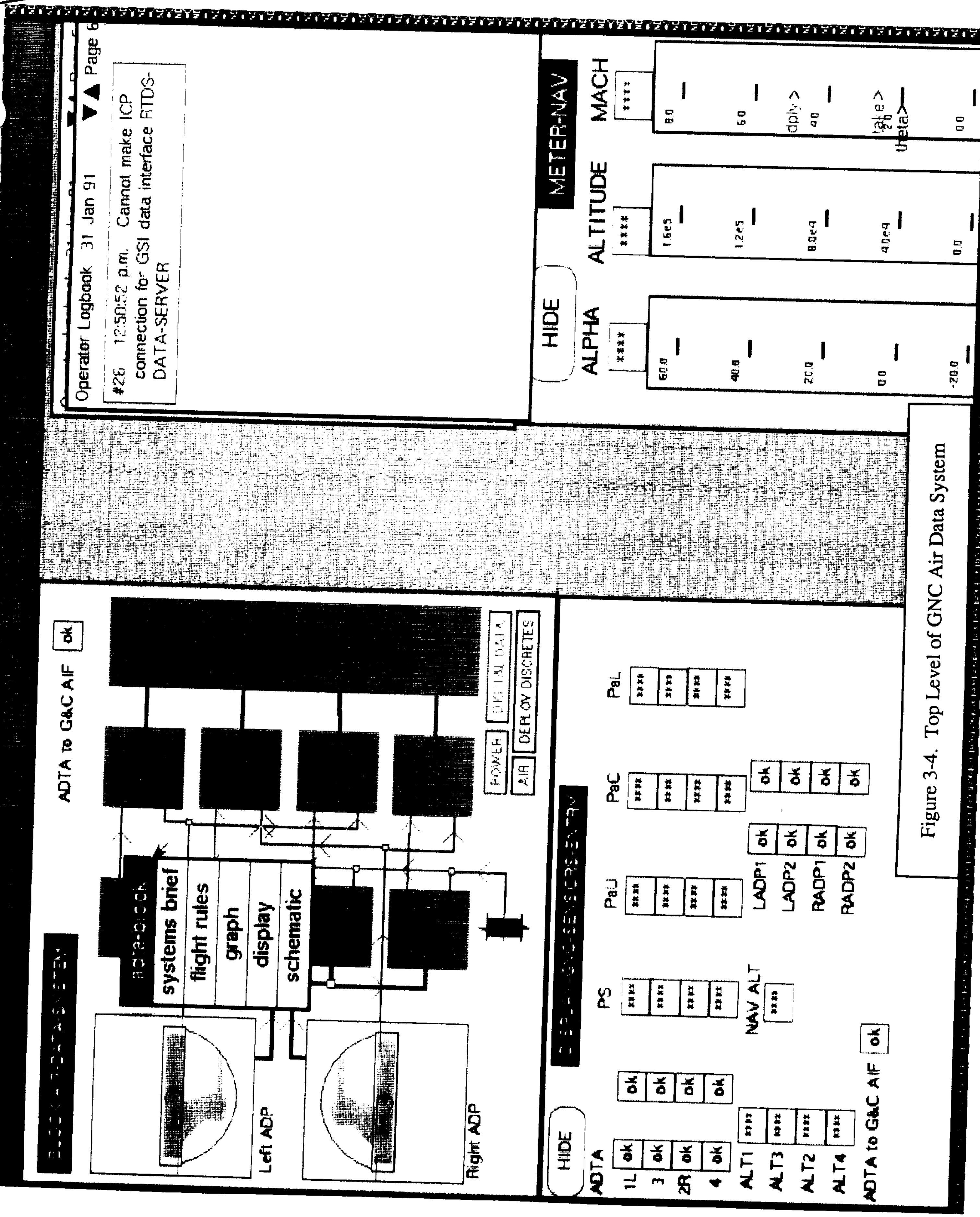


Figure 3-4. Top Level of GNC Air Data System

ORIGINAL PAGE
COLOR PHOTOGRAPH

The second level of the display hierarchy is activated from the schematic. Mouse selection of a subsystem of the schematic (see ADTA block in figure 3-4) displays a menu with the following options:

- System Briefing
  A pre-defined text description of the subsystem; see figure 3-5.

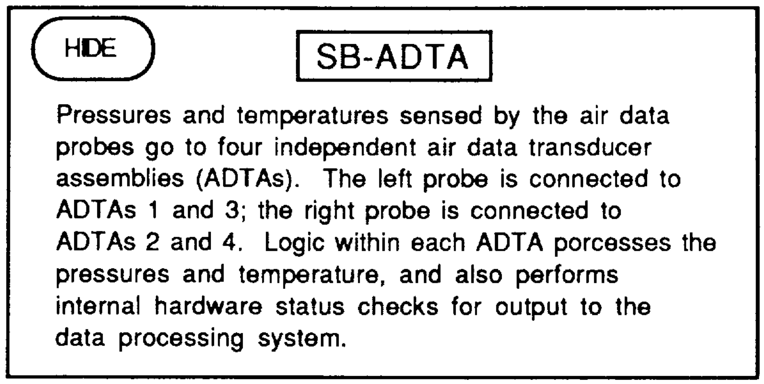


Figure 3-5. System Briefing Option under GNC Air Data System

- Flight Rules
  A hypermedia-like representation of related flight rules. A "rationale" button is also available for access to an explanation of the flight rule. Flight rules are included as antecedents in rules generating recommended actions, but the representation of flight rules in the rule base is not tied to the information base association with the Flight Rules menu option. See figure 3-6 for an example of the Flight Rules option.

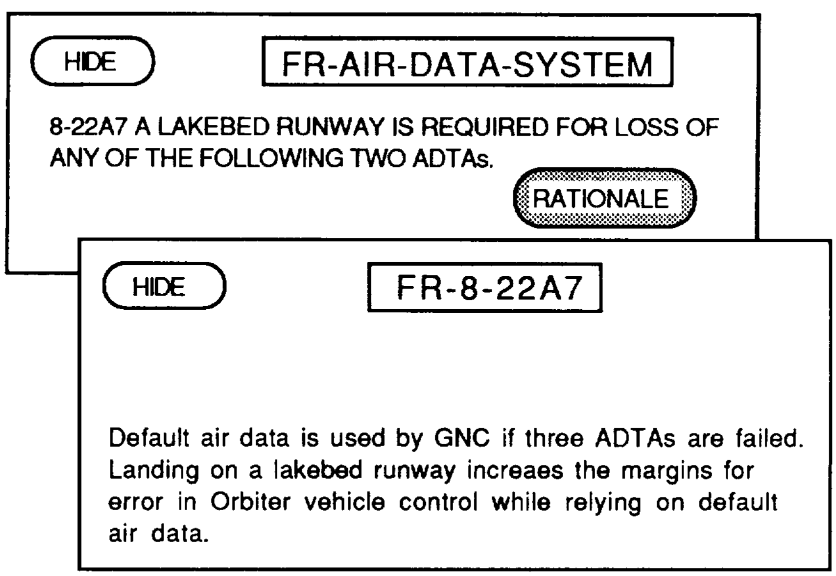


Figure 3-6. Flight Rule Option under GNC Air Data System

- Graph
  A plot of a telemetry parameter versus time that is updated in real-time. The last 2 minutes of data are displayed for the specified MSID. See figure 3-7 for an example of the Graph option.

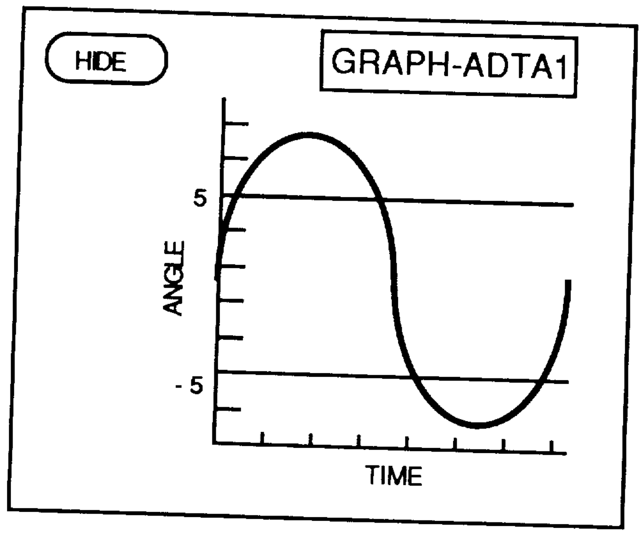


Figure 3-7. Graph Option under GNC Air Data System

- Display
  A tabular display of raw telemetry values; see figure 3-8.

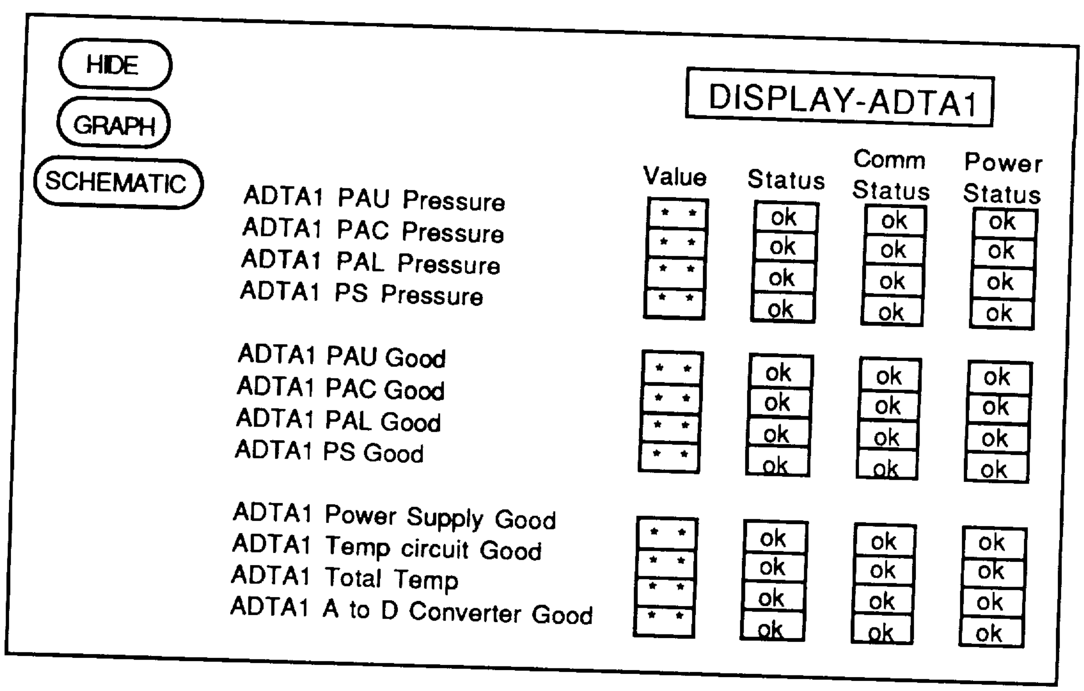


Figure 3-8. Tabular Display Option under GNC Air Data System

- Schematic
  A detailed schematic of the selected subsystem. Components within this schematic are also mouse-sensitive. Selecting a component calls up a table of information about the associated telemetry for that component. See figure 3-9 for an example of the Schematic option.

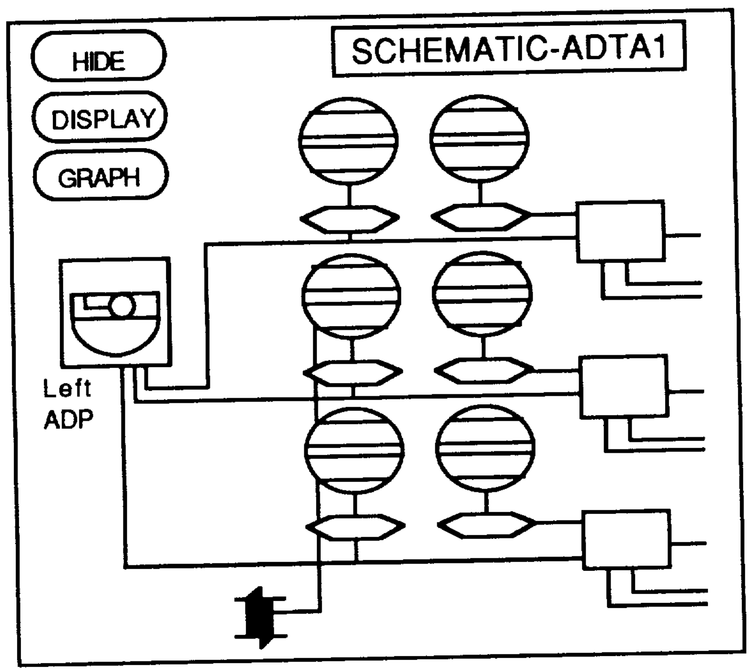


Figure 3-9. Schematic Option under GNC Air Data System

When an option has been selected, a window containing the requested information is provided. Buttons on that window allow direct access to other menu items from the second display level (e.g., the Schematic option can be selected from the window displaying Graph) or return to the previous display level.

Loss of Control

Normally, an operator log book window and two similar windows, one for pitch and one for yaw, are displayed. The operator log book provides a chronological log of each status message generated (see the description of Air Data Probe for more information about the log book). Each angle window has a colored border that indicates the current assessment of LOC:

- Blue
  Angle rates are within limits
- Yellow
  Angle rates are greater than 3 degrees per second
- Red
  Angle rates have been greater than 5 degrees per second for at least 5 seconds

Within each window, two values are displayed in real-time: angle rate and time elapsed since the angle rate limit was exceeded. Two buttons provide access to more information: GRAPH and METER. Selection of the GRAPH button pops up a window containing a plot of angle rate versus time, similar to the graph option in air data probe, with the 5 degrees per second threshold clearly indicated. Selection of the METER button pops up a window containing a sliding scale display of the current angle rate, similar to the flight profile in air data probe, with the 5 degrees per second threshold marked as well. Figure 3-10 illustrates the LOC display.

3.5 Design Process

A variety of development approaches have been used within the GNC area. All applications had active user involvement in the development of the system and rapid, iterative prototyping was the common development process. The GNC prototypes were in various stages of development at the time of the interview, although most of the capabilities were scheduled for operational testing during mid 1990 (with the exception of the Air Data System). The applications that were developed by flight controllers were accomplished as a part-time effort (i.e., Air Data System, approximately 7 months part-time and LOC, approximately 1 month part-time). The applications that were developed by software programmers were developed as a full-time effort (i.e., GNC Jet Control, approximately 1 month full-time). None of these systems had been formally certified at the time of the interview.

The GNC Real-time Monitor was specified by GNC flight controllers and implemented by Ron Montgomery (Rockwell Shuttle Operations Company, RSOC), a software programmer. Most displays were prototyped, tested and used, then modified. This application is currently in use during both training simulations and mission support.

The GNC Jet Control application was specified by Brad Schoenbauer (RSOC), who is a GNC flight controller, and implemented by Ron Montgomery (RSOC). Some performance limitations were encountered with G2. Quite a bit of the development time was spent trying to minimize the number of rules and optimize the performance. This application was selected because the telemetry values do not change frequently which relaxed real-time performance constraints.

The Air Data System was specified and developed by Dave Miller (RSOC), who is a GNC flight controller, and Troy Heindel (NASA-JSC). It was designed to be a prototype to test the ease of having flight controllers develop their own software using G2. Dave felt that probably the most difficult aspect of this process was learning the expert system concepts, since he had no background in this area.

The Loss of Control application was developed by Sedra Walton (RSOC), who is also a GNC flight controller. This application had not been used for flight support at the time of the interview, but GNC planned to use it for the first time during STS-31.

3.6 Study Method

All information about the GNC applications was obtained by interview of the project representatives and demonstration of the prototype on April 18, 1990. The project representative Dave Miller is both a flight controller and a system developer and Ron Montgomery is a system developer.

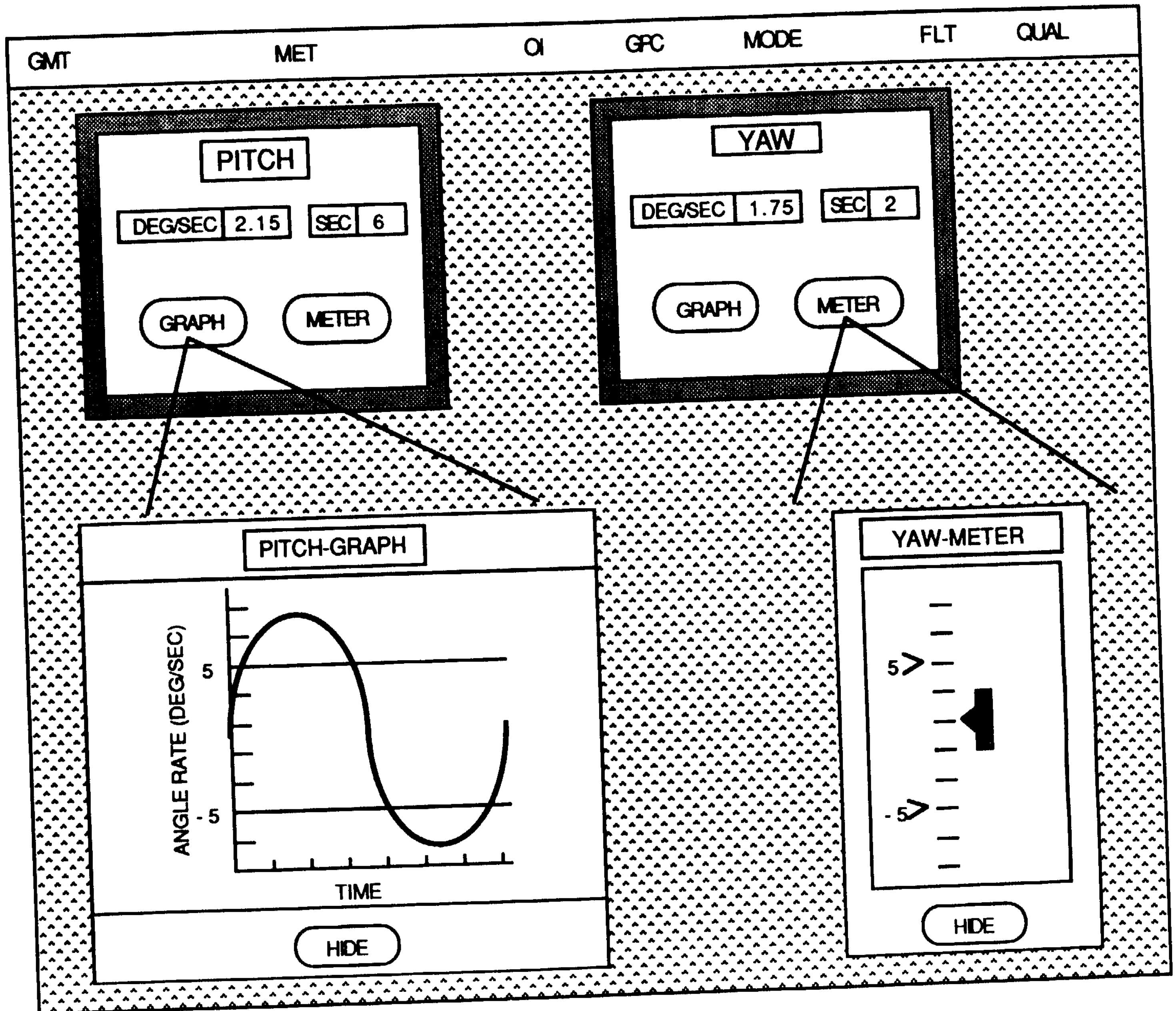


Figure 3-10. Loss of Control Displays developed in G2

Study Team

- Debra Schreckenghost (The MITRE Corporation)

Project Representatives

- Dave Miller (Rockwell Shuttle Operations Company)
- Ron Montgomery (Rockwell Shuttle Operations Company)

3.7 Case Data Sources

No written information was available for the GNC applications. Display hardcopies used throughout this section were provided by project representatives.

Section 4
Instrumentation and Communications Officer (INCO) Expert System Project (IESP)

4.1 System Description

The INCO Expert System Project (IESP) was developed to support the INCO flight controllers who monitor the Space Shuttle communications and tracking systems. The Payload Expert System within the IESP assists the INCO flight controller in fault management of the command and telemetry transmission paths between the payload and the orbiter. Both hard-wired and radio frequency paths are monitored. This expert system also provides improved access to some of the information in the telemetry data over the current display format (e.g., frequency information is provided that currently is only available in Binary Coded Decimal (BCD) format).

The knowledge-based system is built in CLIPS. The user interface is built using a custom-built display interface library with Masscomp graphics. The custom library was built to enhance portability. A port to either the X Window System™ or G2® is planned later. IESP was designed to run on multiple platforms, including the Masscomp 5600 and 6600 and a PC. Most of the data used by the expert system comes from an application called COMPS which performs algorithmic computations on downlisted telemetry.

4.2 Intelligent System and Functions

The Payload Expert System is a rule-based system that provides status and state information about the components in the data communications paths between the payload and the orbiter. This information is used for fault management by the flight controller. The intelligent system both interprets data and provides improved access to data over the current display format. It generates messages that identify important events and system failures.

An interesting observation from the use of IESP was that approximately ninety percent of the problems detected by the expert system are actually misconfigurations. This is because the Space Shuttle flight system is dependent upon manual reconfiguration (i.e., humans are prone to error in high workload environments such as space) and limited telemetry is available for ground operators to distinguish between malfunctions and misconfigurations. The typical reconfiguration procedure after a fault occurs is to bypass failed components, using instead redundant components by selecting an alternate path through the system (i.e., a procedure termed "restringing"). It has been observed that immediately after switching to a new path, the components in that path are usually not in a configuration compatible with the current configuration. This misconfiguration can result in additional indications of a fault.

The Payload Expert System is viewed as a prototype for testing new design concepts for software supporting the INCO flight position. IESP is also considered to be an important form of corporate knowledge capture.

™ The X Window System is a trademark of MIT
® G2 is a registered trademark of Gensym Corporation.

4.3 Human-Intelligent System Interaction Functions

The Payload Expert System is a real-time intelligent system for monitoring and assessment of the state and status of components in-line with the command and telemetry transmission paths between the payload and the orbiter. It provides both interpretation of information and improved presentation of telemetry data. State and status are presented in a variety of ways, including a schematic of the transmission path, message lists, and tabulated information. Messages are assigned priorities to assist the operator in determining whether to interrupt his current activity in response to in-coming information.

The Payload Expert System supports two forms of collaboration, (1) pre-defined text descriptions about specific components of the monitored process and (2) review of both playback data and intelligent system messages (a history of fault messages and a scrollable message window).

For one component (the Controller Interface Unit (CIU) payload link switch), the intelligent system predicts the expected state of the component and compares that to the inferred actual state to detect misconfigurations.

No capability to intervene or control the Space Shuttle communications systems is provided by the intelligent system.

Enhancements planned for future implementation include:

- Use of mission context to bring information to the foreground, such as the display of messages
- Explanation by providing context-sensitive help information and pointers to additional sources of information
- Indication of fault history on the schematic, such that systems previously failed or with previous faulty behavior are identified
- Inferencing approach to controlling what should be brought to the operator's attention. When possible the system would assess the importance of a message or status and, if it is not important, it would not be enunciated. This would be especially useful in fault situations where multiple alarms are generated, many of them redundant or incorrect.

4.4 Supporting User Interface Capabilities

The Payload Expert System has a single screen interface (see figure 4-1). This user interface is constrained to be tiled, with a minimum of popup windows. At the top of the screen is a bar indicating current time and status of data acquisition. In the upper right portion of the screen, two schematics are provided: command paths and telemetry paths. The status of each component and the connectors between are indicated in color, with redundant text coding for components. Color codes in the schematics are:

- Good
  White letters on solid green background
- Potential problem (i.e., a fault or misconfiguration)
  Yellow letters on black background

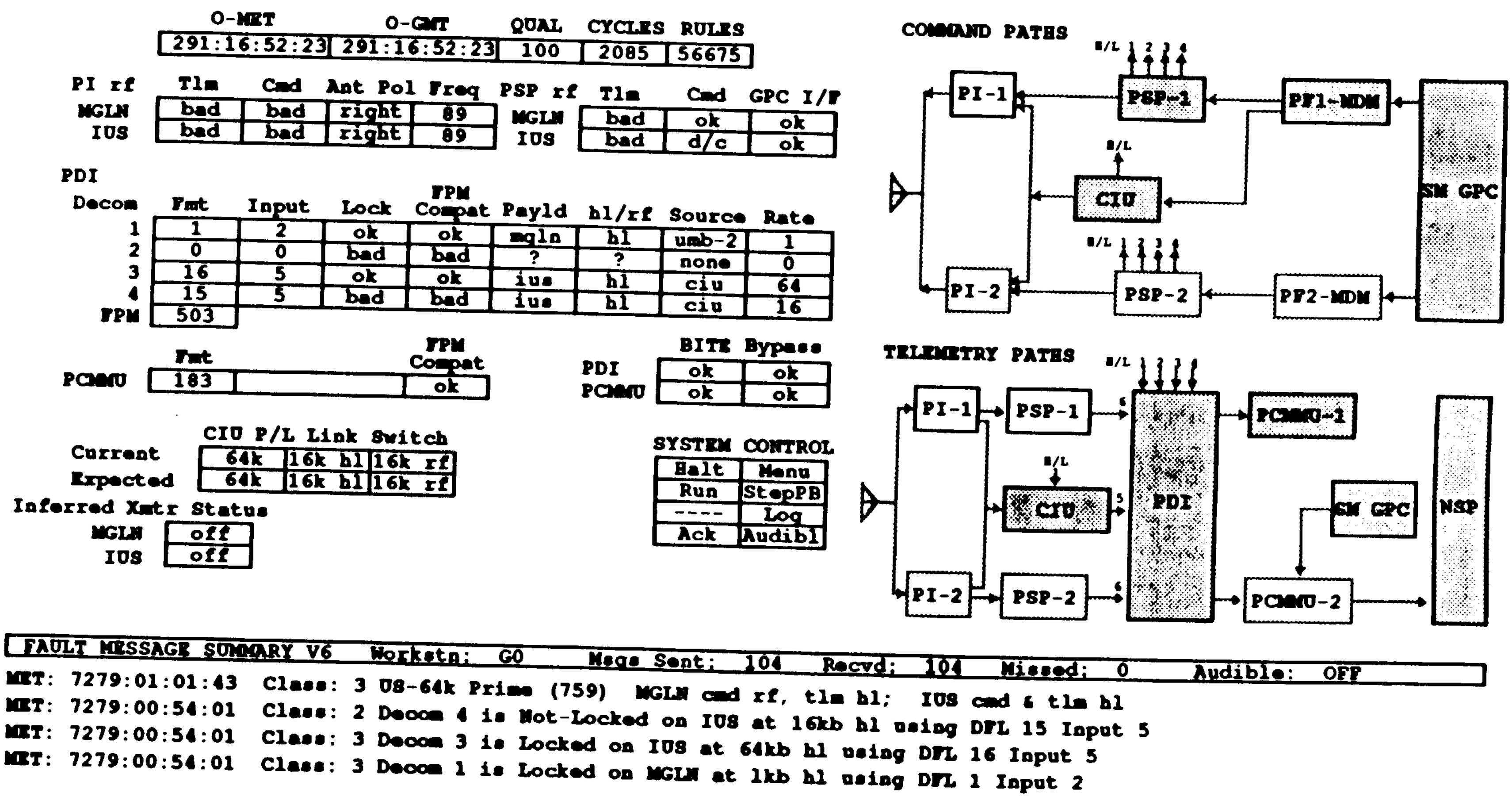


Figure 4-1. User Interface for the Payload Expert System (Rasmussen et al., 1990)

- Problem exists, faulty
  Yellow letters on red background

- Inactive component that had a problem when last active
  (illustrates component fault history)
  Red letters on black background

- Inactive, powered off, or irrelevant to the current configuration
  Grey letters on black background

All components are mouse-sensitive. Using the mouse to select a component provides the following information:

- Help about the component in the form of a pre-defined text description

- Table of data values associated with the component. Two tables are possible, one for telemetry data and one for data from COMPS. The telemetry table displays parameter identifier (i.e., Measurement Stimulus Identification), descriptor string, and current value. The COMPS table displays COMPS name and current value.

To the left of the schematics are a series of panels providing information on each of the components in the communication path between the payload and the orbiter. This panel format loosely emulates the current Digital Display Devices (DDD) format. This information is redundantly coded, with text values displayed in color. Color codes are the same as those used for the schematics, with once addition. White text on a black background is used to designate the best guess for the state of Controller Interface Unit (CIU) switch. Additional information about the CIU is provided below.

In these panels, data from different sources are mixed (i.e., telemetry, data from COMPS, and data inferred by expert system) and source is not distinguished on the display. Some values from COMPS are determined by limit checking. These limits are mission specific and do not change for the duration of the mission. Information displayed on the panels includes:

- Status of the data acquisition and expert system software
  1) QUAL: indicates number of successful frames of data (0-100) from the previous 100 data acquisition frames
  2) CYCLES: total number of major cycles of the rule base
  3) RULES: total number of rules fired

- Key information characterizing the major components of the communication paths

  1) Payload Interrogator (PI): Evaluation of the health of radio frequency components in both command and telemetry path for each payload, the antenna polarity assessment, and the inferred frequency

  2) Payload Signal Processor (PSP): Evaluation of health of radio frequency components in both command and telemetry path for each payload and an assessment of General Purpose Computer (GPC) interface (inferred from GPC status)

  3) Payload Data Interleaver (PDI): Telemetry (format, input, lock status) and inferred information (format compatibility, payload identifier, type of path, source, and rate) for each of 4 available channels

4) Pulse Code Modulation Master Unit (PCMMU): format number and assessment of format compatibility

5) CIU: The expert system provides a best guess of the probable state of the CIU payload link switch based on evidence in the telemetry. It also infers the expected state of the switch. One or more states are selected by the expert system from the possible values for both the probable state and the expected state. The "best guess" is displayed on a panel in white text while the remaining values are displayed in grey text. Mismatches between what is expected and what is concluded as the probable state indicate a likely misconfiguration. Previously, flight controllers were forced to contact the crew and ask the state of switch.

6) Built-In Test Equipment (BITE) and bypass status for PDI and PCMMU

7) Inferred status of transmitter (on or off) for each payload

At the center of the screen is the expert system control table. It provides the following user capability for the Payload Expert System:

- Start and halt the expert system
- Control data gathering by suspending or enabling the system
- Acknowledge a fault message
- Step through playback data
- Enable and disable logging of data
- Enable and disable audible fault warnings

The bottom portion of the screen contains a time-sorted list of all messages from the various IESP applications. All new messages are displayed in inverse video and an audible alert is optionally issued. The controller can acknowledge new messages using the system control panel, which results in display of the message in normal video. The text of messages is color-coded:

- Grey
  Advisories (e.g., state transitions)
- Yellow
  Warnings (e.g., misconfiguration)
- Red
  Major problems or failures requiring immediate attention

These messages are sorted chronologically and are scrollable. Each message entry consists of time, priority assessment of message, and content of message. The total area occupied by the fault message portion varies for different IESP applications. For the Payload Expert System is occupies approximately one third of the screen. A Fault Message Review screen is also available to examine the fault message history log. Messages from all IESP applications are interleaved and time-sorted.

4.5 Design Process

The initial requirements for IESP were defined in working meetings between the intelligent system developer and an INCO flight controller. From the beginning, IESP was defined in an electronic media instead of the more typical paper requirements definition. No formal specifications were written and only informal studies were performed. A few paragraphs of high level requirements resulted from these meetings. The goal of the initial effort was to develop a demonstrable prototype within 3 months. The initial prototype was refined iteratively, with frequent user evaluation. Ideas were quickly evaluated in software to determine if they were useful. If not, the effort was immediately re-scoped. This lead to a flexible software development process of iterative upgrades. An upgrade consisted of adding new features to determine if they enhanced the new capability in the "right direction". Each upgrade had a specific, short-term target date for completion and demonstration which kept the operators in the requirements definition and evaluation process. IESP went through 3 or 4 generations of upgrade, with each successive upgrade involving fewer changes with less impact. The first user interface was an advanced display with considerable use of new interface concepts. It was well-accepted by the operator involved in the initial design, but was considered by the other operators as too different from existing operational displays. This initial wrong focus resulted in a major redesign to a system that more closely resembled the existing interface. Remaining redesigns tuned the previous designs. The interface design process included evaluation of operations scenarios to assist in identifying what is important early in the project and to avoid perpetuating an incorrect or improper emphasis. Evolutionary issues include scalability of small-scale prototypes to large operational systems and ability to certify prototypes.

The certification process was described as one of side-by-side comparison between the existing support system and the new support system during operations (i.e., training simulations, missions). The system is formally certified when a designated member of the operations signs off on the capability. The IESP expert system is not yet certified.

4.6 Study Method

Information about the IESP was obtained by interview of the project representative on March 2 and demonstration of the prototype on May 2, 1990, and by review of the case data sources cited below. The project representative Art Rasmussen is the system developer.

Study Team

- Debra Schreckenghost (The MITRE Corporation)

Project Representative

- Art Rasmussen (The MITRE Corporation)

4.7 Case Data Sources

Brown, Daryl and Tom Kalvelage (September, 1988), *INCO Expert System Project Real Time Data System User's Guide*, Mission Operations Directorate, Johnson Space Center, Houston, TX: NASA.

Muratore, J.F., T.A. Heindel, T.B. Murphy, A.N. Rasmussen, and R.Z. McFarland (December, 1990), "Real-Time Data Acquisition at Mission Control", *Communications of the ACM*, Volume 33, Number 12, pp 18 - 31.

Rasmussen, A.N., J.F. Muratore, T.A. Heindel (August, 1990), "The INCO Expert System Project: CLIPS in Shuttle Mission Control", *Proceedings of First CLIPS Conference*, NCP 10049, Volume 1, Johnson Space Center, Houston, TX: NASA.

Section 5
KU Band Self-Test Expert System

5.1 System Description

The KU Band Self-Test Expert System is an application built for use by the Instrumentation and Communication Officer (INCO) during flight support. This application assists flight controllers in monitoring the checkout, or *self-test*, of the KU band radar located onboard the Space Shuttle. This test is performed once per mission and lasts for 220 seconds. The self-test consists of the sequential execution of sub-tests called data blocks that are assessed as either passing or failing. If a data block passes, execution of the next data block occurs. If a data block fails, either execution branches to a data block further in the sequence or the test fails completely. Currently, INCO flight controllers must monitor portions of two displays to access the information needed to monitor the self-test. This application provides improved access to necessary information during the test as well as allowing playback of data after the test for review of the test sequence. Data are logged to file during the test to permit playback.

The KU Band Self-Test Expert System was specified and developed by an INCO flight controller (George Pohle). No formal requirements were specified. The application was developed in G2® and he had no G2 training prior to this application. The application resides on a DEC® 3100 workstation. The system is approximately 30% complete. Three data blocks have been implemented for a total of 200 rules. It was estimated that the majority of these rules were for display instead of monitoring functions. A simulation has also been developed as an alternate source of data for testing. Efforts at the time of the interview (July, 1990) were focused on interfacing to real data via the Real Time Data System (RTDS) telemetry distribution system. Other flight controllers have seen the application and provided feedback. This system is planned for flight support, but has not yet been used operationally.

A simulation was created to allow the system developer to generate data representing a nominal test sequence or to selectively fail data blocks via parameter changes during selected time periods. This simulation capability was provided for testing of the program during development. Once the system has been certified, the simulation can be used to generate test cases for training of flight controllers using this application. Performance problems have been seen in executing the application with the simulation running. Figure 5-1 provides an example of the simulation workspace.

® - G2 is a registered trademark of Gensym Corporation

® - DEC is a registered trademark of Digital Equipment Corporation

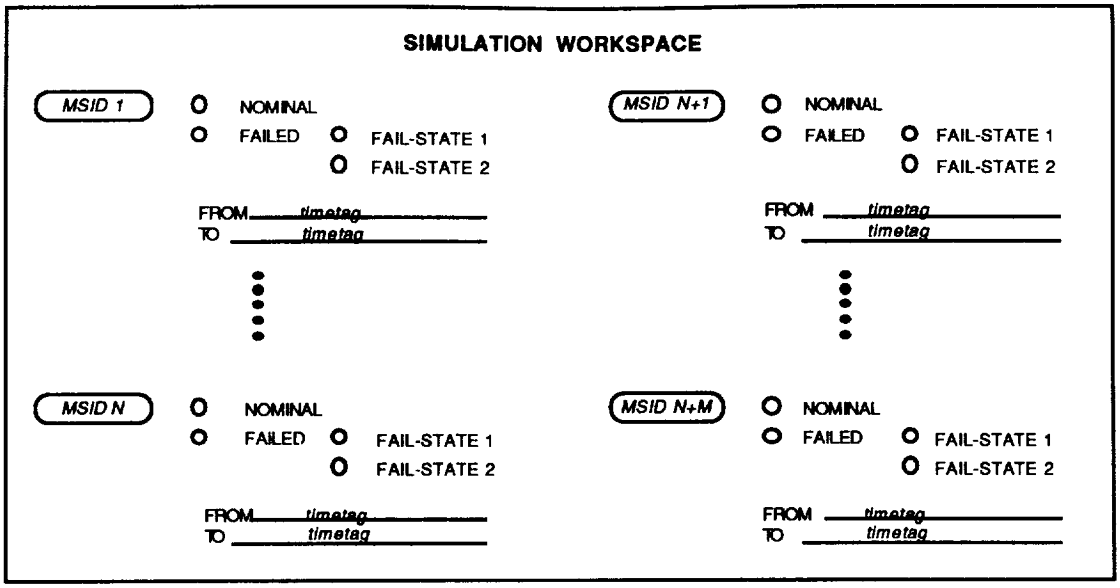


Figure 5-1. Simulation Workspace

5.2 Intelligent System and Functions

The KU Self-test Expert System is a rule-based system. It is designed to assist the operator in performing a self-test of the KU Band radar and in assessing the results of that self-test. It orients the operator about position within the self-test and indicates how well the self-test is going. The self-test consists of an ordered series of sub-tests, called data blocks. The intelligent system identifies which data block in the self-test is currently being executed and, once the data block execution is complete, assesses the status of that execution (i.e., pass for nominal behavior, fail for anomalous behavior). The assessment of data block execution status involves monitoring parameters affected by data block execution to determine if their value indicates nominal or anomalous behavior. The intelligent system provides both the expected parameter changes resulting from the data block (assuming that the data block passes) and the actual parameter changes after data block execution. The intelligent system also identifies the next expected data block, based on the status of the current data block execution. The operator can optionally access additional information about the self-test, including an explanation of the status of the current data block and the expected results (i.e., time of transition to next data block and parameter changes) from executing the next data block (assuming it passes).

5.3 Human-Intelligent System Interaction Functions

The primary function of the intelligent system is procedure monitoring and assessment. It locates the current data block activity within the overall checkout of the radar and assesses the success of each data block activity. The intelligent system can also predict the expected effects of the next data block based on nominal execution of that data block. Information are presented in the form of message lists and tables.

Collaboration is supported in two ways: (1) explanation and (2) data playback and review. Explanation consists of pre-defined text blocks associated with specific data blocks. Data playback of the self-test allows the operator to review the results of selected data blocks much slower than real time. Any of the available displays may be viewed during review.

The intelligent system is a passive monitor only. It provides no capability to intervene with or control the execution of the self-test.

Two explicit modes of operation using the intelligent system are planned, although only one (VERBOSE) had been implemented at the time of the interview:

- TERSE
  The minimum amount of information is displayed to determine if the data block passed or failed (figure 5-2 in section 5.4)

- VERBOSE
  Detailed information from a data block can be monitored; this mode has been implemented (figure 5-4 in section 5.4)

The prescribed use of these modes is performing normal operations in TERSE mode, with playback in VERBOSE mode after a data block failure to review what went wrong. The VERBOSE display would provide all information on the TERSE display with additional windows for specific data block information. Note that throughout all displays the downlisted telemetry parameters are displayed with a descriptive name instead of the Measurement Stimulation Identification (MSID) number which is typically used to identify downlisted parameters (e.g., MODE, not V12X3456).

Although the KU Self-test Expert System was designed for real-time support, it has potential application in training flight controllers. The simulation used to create test cases can be used to create training cases for an off-line trainer.

5.4 Supporting User Interface Capabilities

Four windows are displayed on the TERSE display (figure 5-2):

- Time and Mission Configuration
  Displays times and software configuration

- Window Select Buttons
  Select button to display "Show All Data" window or "Self-Test Flags" window

- "Show All Data" Window
  Displays current values of downlisted parameters

- "Self-Test Flags" Window
  Displays current status of data block execution

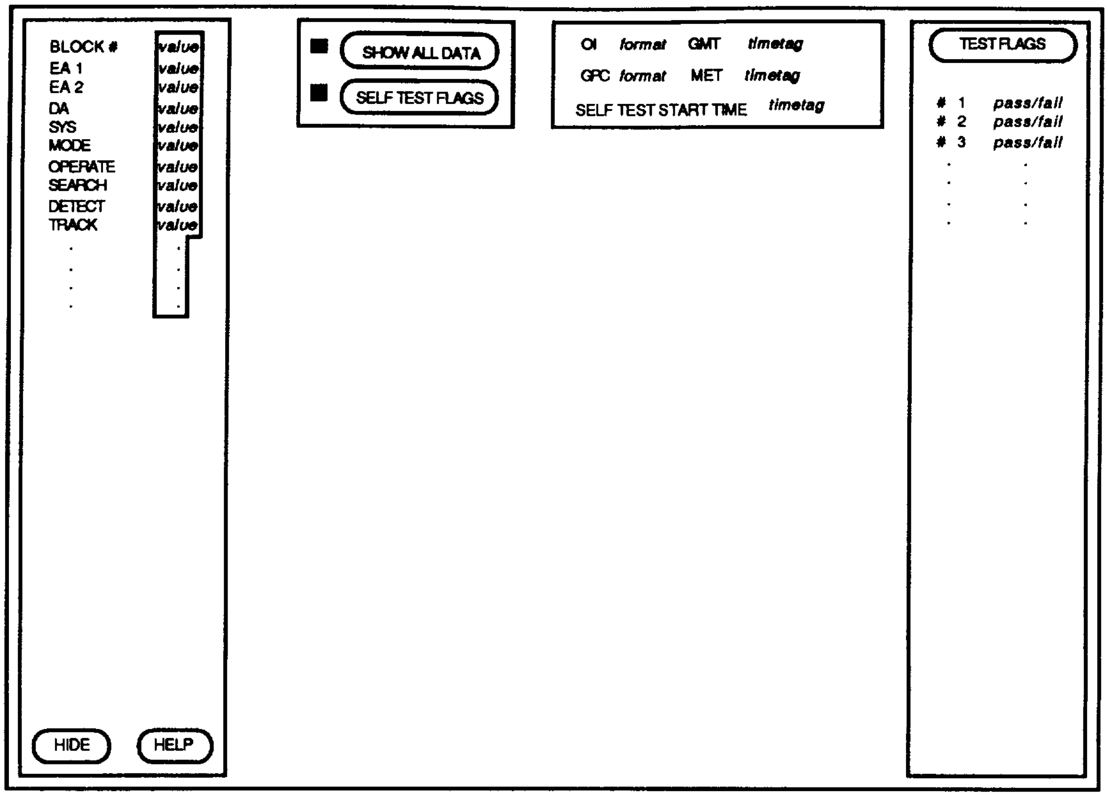


Figure 5-2. KU Band Self-Test TERSE Mode Display

The labels in the "Self-Test Flags" and "Show All Data" windows are color-coded. Only the labels change color, however. The values associated with the labels are displayed in black text. For the "Self-Test Flags" window, color is used to indicate status of a data block. A green block number indicates nominal status during data block execution and a red block number indicates off-nominal status during data block execution. For the "Show All Data" window, the color coding indicates both status and expected sequence of executing data blocks. Since a MSID may be changed by more than one data block, MSIDs expected to change in the next data block are differentiated from MSIDs expected to change in the current data block. The color coding of labels reflects the current data block. Color conventions for "Show All Data" window are:

- Green
  Data value nominal (i.e., passed test) for current data block and only expected to change in current data block
- Yellow
  Data value is set by test, but value should not change during current or next data block
- Purple
  Data value changed during current data block and is expected to change during next data block as well
- White
  Data value not expected to change during current data block but should change during next data block (i.e., "next event" parameters)

- Red
  Data value off-nominal (i.e., failed test) for current data block
- Black
  No activity expected in data value

Alternate formats for the "Show All Data" window are under investigation. Figure 5-3 shows one possible alternative. The motivation behind a horizontal display of data is to accumulate data values into tabular form, with values from the current data block on the top row and values from previous data blocks sorted by time on the rows below. Previous data values would be accessible by scrolling.

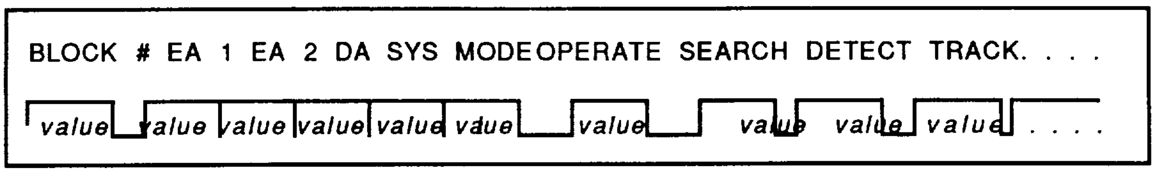


Figure 5-3. Horizontal Format for "Show All Data" Window

The VERBOSE display adds a series of four popup windows in the center of the screen to the display format of TERSE mode. These windows provide information on the current data block and the next expected data block. A planned enhancement is to clearly indicate data block execution sequence by displaying previous, current, and next data blocks simultaneously. These windows as displayed from top to bottom are:

- Execution Times
- Data Block Analysis
- Next Data Block
- Explanation of Current Data Block

Each of these popups is discussed below. See figure 5-4 for an illustration of the VERBOSE display.

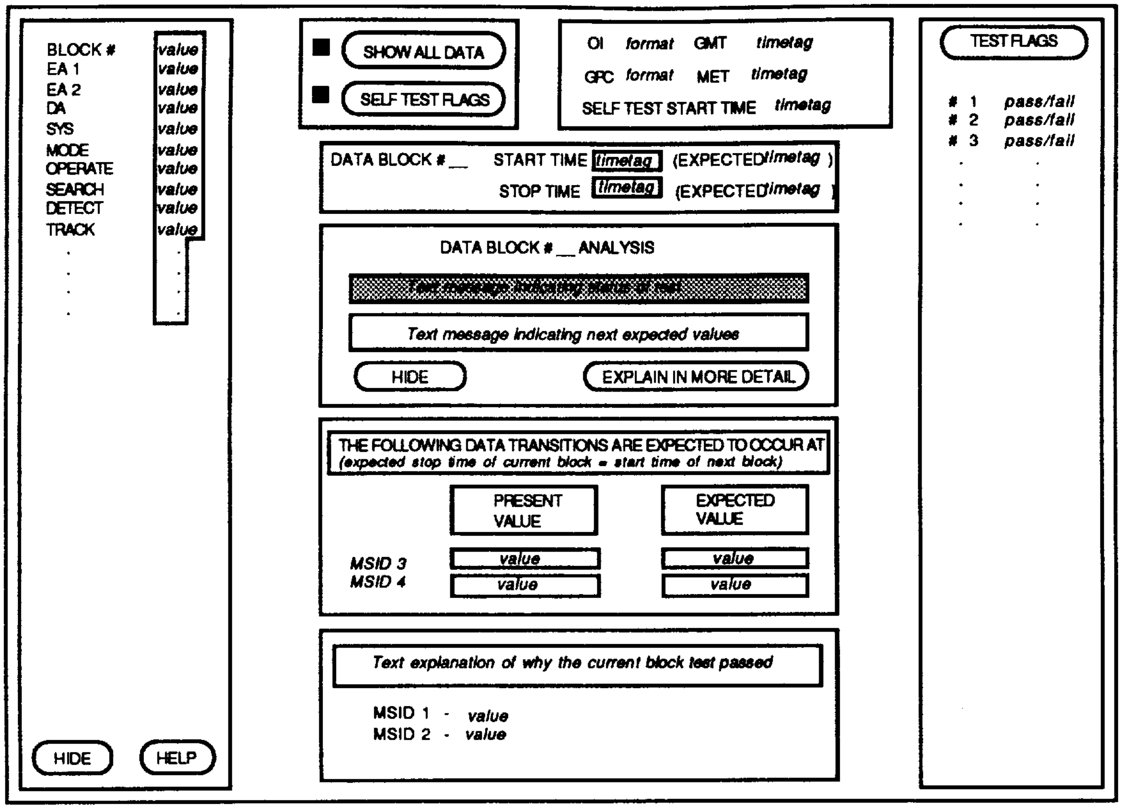


Figure 5-4. KU Band Self-Test VERBOSE Mode Display

When in VERBOSE mode, the operator cannot hide the detailed displays until the transition to the next data block. Currently, these displays layer on top of each other unless the operator manually hides old displays. Possible enhancements mentioned were automatically hiding old displays or hiding one detailed window to hide all detailed windows associated with the current data block.

Figure 5-5 shows the first popup window, Execution Times. This window displays the start and stop times of a data block. The times displayed in the boxes to the right of the START and STOP labels are the actual start and stop times while the times displayed in the field to the right of the EXPECTED label are the expected start and stop times (i.e., values from mission-specific initialization load or I-load). The intelligent system assesses the status of the data block (i.e., pass or fail) when the actual stop time occurs.

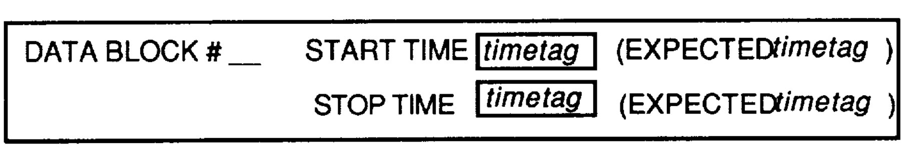


Figure 5-5. Execution Times, First Popup Window of VERBOSE Display

The second popup window of the VERBOSE display provides an analysis of the data block currently being executed. Information displayed includes a text description of the status of the data block execution (i.e., nominal or off-nominal) as determined by the intelligent system and the expected MSID values resulting from the execution of the current data block. The box around the status of the data block execution is color-coded as well, with green for nominal and red for off-nominal. Two control buttons are available:

- HIDE
  Hides the workspace
- EXPLAIN IN MORE DETAIL
  Creates a popup window with additional information about the data block; explanation consists of pre-defined text descriptions

See figure 5-6 for an example of the Data Block Analysis window.

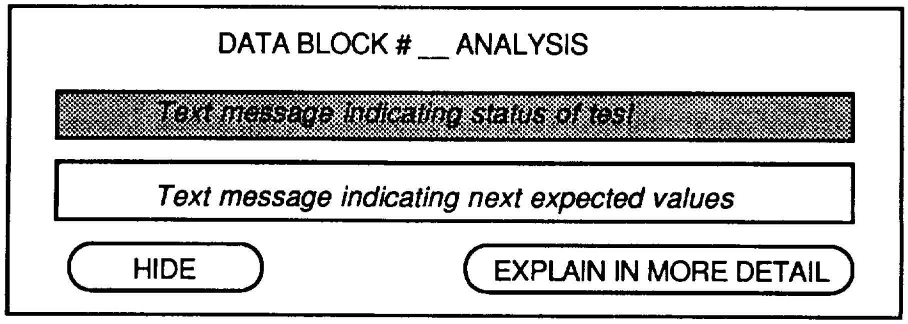


Figure 5-6. Data Block Analysis, Second Popup Window of VERBOSE Display

The third popup window of the VERBOSE display is the operator's view into the expected effects of the next data block. The window includes the expected time of transition to the next data block and both the present and expected values for all parameters affected by the next data block. See figure 5-7 for an illustration of the Next Data Block window.

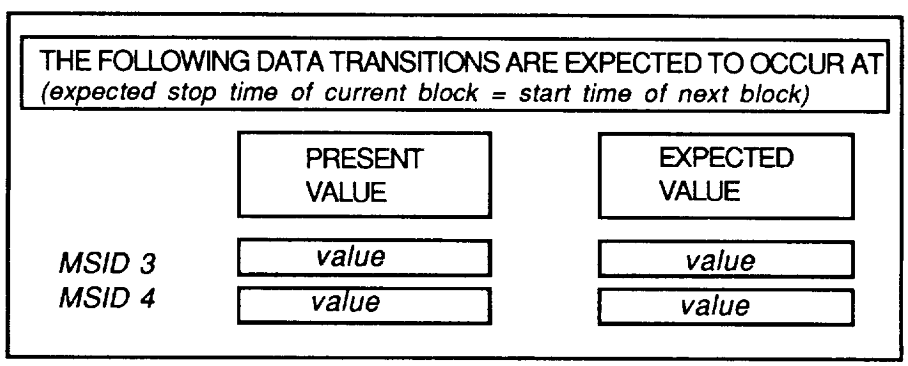


Figure 5-7. Next Data Block, Third Popup Window of VERBOSE Display

The fourth and final popup window of the VERBOSE display provides an explanation of the status of the current data block that is displayed in the second popup window of the VERBOSE display (i.e., Data Block Analysis window). This explanation consists of a text message and a list of the values of all MSIDs relevant to the current data block. See figure 5-8 for the format of the window providing an Explanation of Current Data Block.

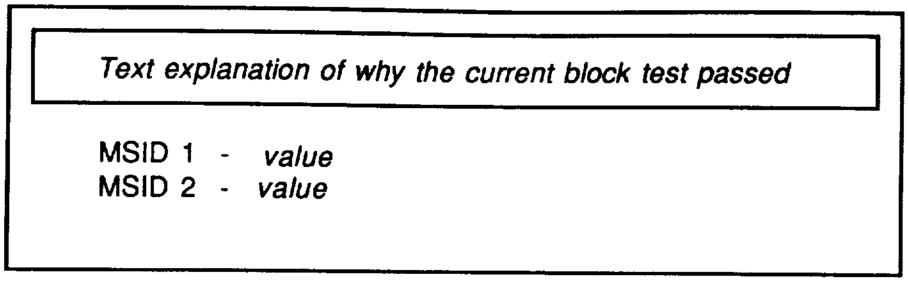


Figure 5-8. Explanation of Current Data Block, Fourth Popup Window of VERBOSE Display

5.5 Design Process

The KU Band Self Test Expert System is being developed by Space Shuttle flight controllers. Thus, the user (i.e., flight controller) is also the domain expert and the software developer. Such systems are obviously characterized by early, active user involvement. The development process includes rapid, iterative prototyping with quick deployment into an operations-like environment for side-by-side testing with the existing support displays. Such side-by-side testing has proven effective in integrating the new technology into the existing support environment for other RTDS applications. At the time of the interview (July, 1990), development of the knowledge base was in progress. In addition to providing data from integrated training simulations, a simulation that allows developer control over failures was written for use during testing. The KU Band Self Test Expert System uses the RTDS data acquisition support system.

5.6 Study Method

All information about the KU Band Self Test Expert System was obtained by interview of the project representative and demonstration of the prototype on July 9, 1990. The project representative George Pohle is both a flight controller and a system developer.

Study Team

- Debra Schreckenghost (The MITRE Corporation)

Project Representative

- George Pohle (Rockwell Shuttle Operations Company)

5.7 Case Data Sources

No written information was available for the KU Band Self Test Expert System. Since hardcopies of screens were not available, drawings used throughout this section are based on observations made during the demonstration.

Section 6
DATA COMM Expert System

6.1 System Description

The DATA COMM Expert System is an application built to assist the DATA COMM flight controller in the monitor and control of the Space Shuttle onboard flight data recorders. The primary purpose of this flight position is to perform data recorder management (e.g., what data has been recorded, which tape contains desired data). The ultimate goal of the DATA COMM Expert System is to completely automate the DATA COMM flight position (i.e., replace the human controller). This goal is not likely to be achieved soon, because part of the DATA COMM flight responsibility includes uplinking recorder control commands to the Space Shuttle. Such autonomous command initiation is viewed as risky, since it introduces the potential for erroneous command entry. In the interim, this expert system is planned for use by DATA COMM flight controllers.

The DATA COMM Expert System is one of the applications that uses the Real Time Data System (RTDS) data acquisition support system. It is a rule-based system built in G2®. The application consists of roughly 200 rules and is approximately 80% complete. The demonstration was conducted on a DEC® 3100 workstation. The application was demonstrated by George Pohle, but was not developed by him. The original developer has since left this project and George Pohle has resumed responsibility for it. It has been used a few times during simulations and has been reviewed by most DATA COMM controllers with positive response. Testing of this application was planned to begin soon after the interview (which was held in July, 1990).

6.2 Intelligent System and Functions

The DATA COMM Expert System is a rule-based system. This intelligent system provides a health and status assessment of the components that are required to record and downlist data. These components include the onboard data sources (e.g., signal processors, Closed Circuit TV (CCTV), the flight recorders (i.e., OPS[1] 1, OPS 2, and Payload), command capability, and logging capability. The intelligent system generates two classes of messages: system messages and fault messages. System messages identify events of interest, such as Loss of Signal (LOS). Fault messages identify component failures that affect the ability to record or downlist data.

The intelligent system also assists the operator in managing information about the recording process. Such information includes:

- Location on a tape where specific data are recorded (i.e., track)
- Speed of recording
- Time of recording
- Mode of the recorder (e.g., record, playback, etc)

® - G2 is a registered trademark of Gensym Corporation
® - DEC is a registered trademark of Digital Equipment Corporation
[1] The OPS number associated with the data recorders identifies the software load used onboard Shuttle. OPS numbers correspond to mission phase, where OPS 1 is loaded during ascent and OPS 2 is loaded during on-orbit.

Additionally, the intelligent system distinguishes data recorded during LOS (i.e., when data cannot be downlisted to the ground) from data recorded during Acquisition of Signal (AOS) and identifies regions of noisy data. These information items are used by the operator to schedule data for downlisting to the ground, since data recorded during LOS should be downlisted first, followed by noisy data.

The intelligent system maintains a separate record of the logging activity of each recorder. This log provides a summary of which data has been downlisted. The ability to search for specific data segments by time regions is provided for the purpose of determining if the data have been downlisted or not.

6.3 Human-Intelligent System Interaction Functions

The intelligent system was designed as a real-time monitoring and information management support system for the DATA COMM flight controllers. Schematics and message lists present status assessments about components of the system for recording and downlisting data. Information that describes the recording process are monitored as well. Manipulation capabilities for this information about the recording process include generating customized tables of logging activity and data search by time region.

An extensive review capability is provided in addition to the real-time capability. Messages from the intelligent system are logged to allow later review by the operator. A mechanism to buffer messages is provided to assist the operator in managing interruptions by these messages. An icon representing each message buffer changes appearance when a new message is received. Other information recorded for review includes logging activity, parameters that indicate status of the recording process (e.g., plots of recording head temperature over time or percentage of a tape recorded during LOS), and information that describes how data were recorded (e.g., tables specifying recorder mode, speed, and time a given tape location).

Collaboration with the intelligent system is in the form of pre-defined, explanatory text. An Online Help capability can be accessed for additional information about (1) how to navigate through the user interface, (2) information about the expert system, and (3) reminders and warning for DATA COMM flight controllers.

There is currently no capability to intervene into or control the recording or downlisting process using the intelligent system, although such intervention capability is a planned enhancement for later versions of the intelligent system.

6.4 Supporting User Interface Capabilities

Information on the user interface is organized hierarchically with a control panel at the top level. The control panel is used to select other windows for display. At the top of the display is a fixed region displaying Greenwich Mean Time (GMT), Mission Elapsed Time (MET), and the number of bad data frames from the data acquisition system in the last one hundred cycles (QUAL). A fixed region similar to this is usually on all RTDS user interfaces. Figure 6-1 illustrates the top-level control panel display.

The Control Panel is not the only means of calling up windows in the interface hierarchy. Some windows can be accessed via mouse selection from windows hierarchically above them. Popup windows are not constrained to a tiled layout. Once a window has been displayed, a window manager is available to move the window on the screen.

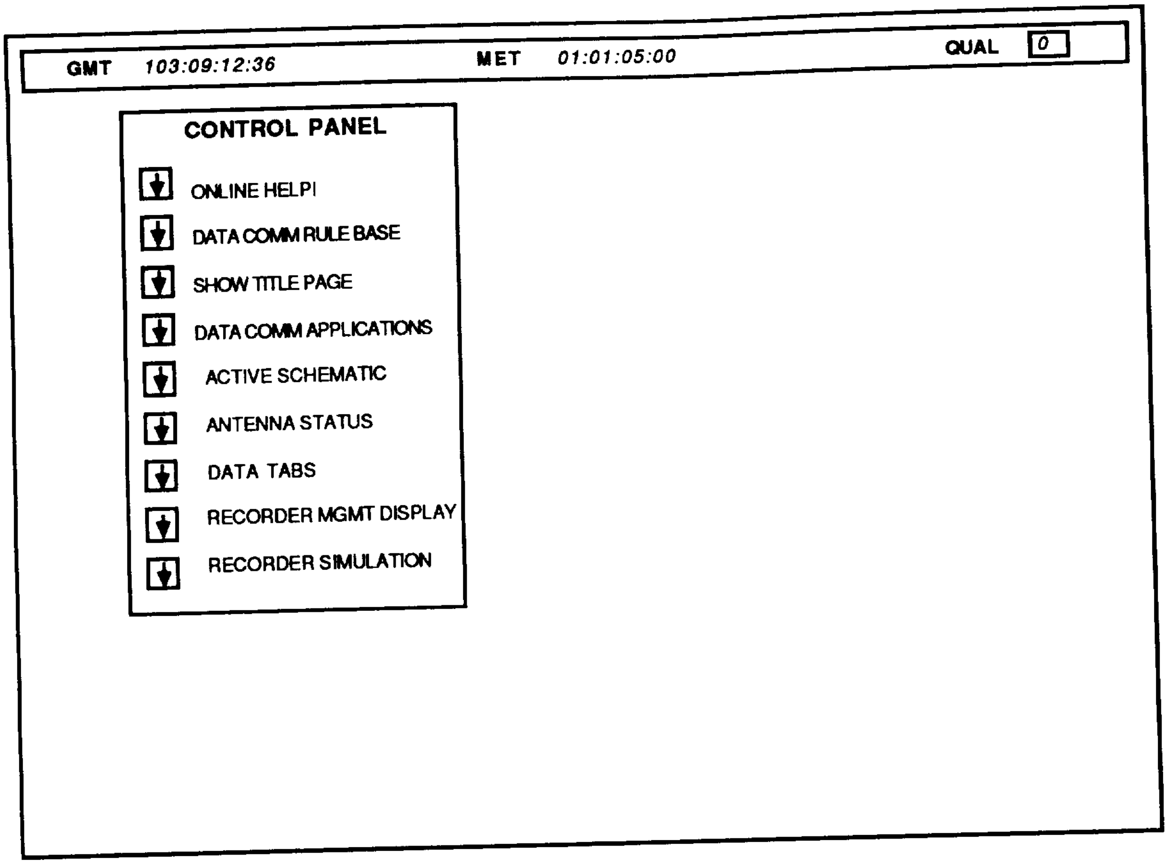


Figure 6-1. Control Panel at Top of Display Hierarchy

Online Help

Selection of the ONLINE HELP! button from the Control Panel pops up another panel of buttons specifying types of help information available. These are:

- Move Around System
  Assistance in navigating through the application interface

- Expert System Information
  Information about the expert system

- DATA COMM Dos and Don'ts
  Reminders and warning for the DATA COMM controller

Specific help options were not demonstrated, but I believe all information is in the form of a pre-defined, hard-coded explanation. See figure 6-2 for an example of this window.

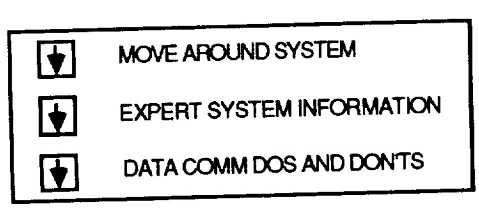


Figure 6-2. Online Help Information Options

Active Schematic

The primary display monitored by the DATA COMM flight controller is the Active Schematic from the Control Panel. This schematic represents the subsystems of the onboard data recording and downlist capability and the paths connecting these subsystems. Primary subsystems include the onboard data sources (e.g., signal processors, CCTV, Payload patch panel), the data recorders (OPS 1, OPS 2, and Payload (PL)), and the radar/antenna assemblies used for data downlist and command uplink. Status of the subsystems is indicated in text boxes either positioned beneath the subsystem icon (e.g., "go" or "no go" for FM, antenna electronics, KU band radar) or collected into a table on one side of the schematic. Status values provided in this table are:

- Data recorder status for OPS 1, OPS 2, and Payload
- Command status (i.e., ability to uplink commands to recorders)
- Log file status

These go/no-go status assessments are determined by monitoring commands to the signal processor. The system merely reports status and does not make a judgement about the specifics of why "no go" was assessed.

Figure 6-3 illustrates the Active Schematic. The graphic in this illustration is representative of the type of graphic employed and does not exactly duplicate the graphic actually used. For example, figure 6-3 illustrates a single path where multiple actual paths may exist.

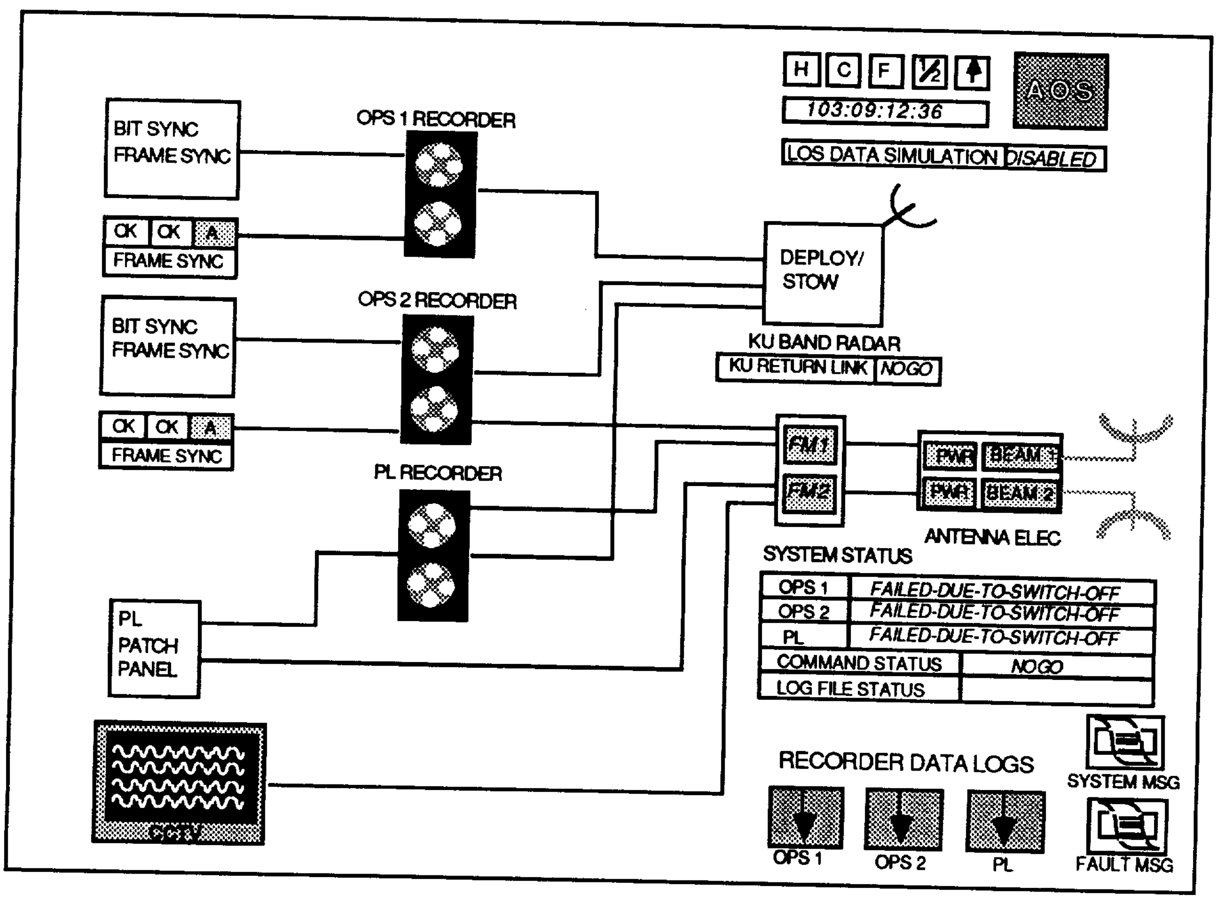


Figure 6-3. Active Schematic for OPS Recorder Management

In the upper right hand portion of the schematic, five display control buttons are provided. These buttons allow hiding or re-sizing the current window or calling up another window. This same panel of buttons is used throughout the interface as a display control mechanism. Figure 6-4 illustrates the display control buttons.

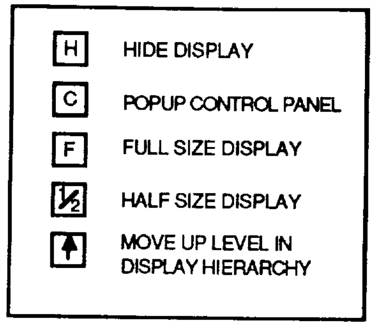


Figure 6-4. Display Control Buttons Provided on All Windows

The icons for the data recorders and the CCTV change dynamically to indicate current state. For the data recorders, the tape icons rotate when data are being recorded. This rotation continues after LOS occurs during a recording period, based on the assumption that nominal operations prior to LOS indicates probable nominal operations during LOS. A planned enhancement is to verify that recording is still nominal when signal is re-acquired (i.e., AOS). The screen of the CCTV icon changes to represent operational activity. Figure 6-3 illustrates a static-filled screen, used when no data are being generated by the CCTV. When data are being generated, the screen shows the Earth horizon with the Sun moving along the limb.

It was not possible to determine all of the color conventions used in this application. From observation, however, a number of conclusions can be made. Color-coding is used to indicate the state of the data recorders, the onboard data sources, the antenna electronics, and the message files. Paths between subsystems show green when active and black or grey when inactive. Grey-tone is also used to indicate disabled status (e.g., grey tone on FM antennas when disabled). The signal acquisition status is shown in a lighted panel on the display. AOS is indicated by a green panel containing the text "AOS". Probably LOS is indicated by red or orange.

The schematic includes icons that are sensitive to mouse selection. Selection of these icons enables the popup of a window containing information related to the subsystem or data item represented by the icon. Sensitive icons include:

- Data Recorders

  Two windows may be accessed from the data recorder icons on the schematic: the recorder management table and the recorder quick look summary. The recorder management table may be called up from the Control Panel as well.

  The recorder management table documents the percentage of tape used versus the track number for each recorder. Arrow heads, pointing to the right, are used as markers to relate a percentage to a track. Additionally, these markers are mouse-sensitive. Selection of a marker calls up a data table similar to an existing data format (i.e., Data Tabs) provided by the Mission Operations Computer (MOC). The source of the display name is the right-shift ("tabbing") of information in the table when a snapshot of current data is stored in the table. The information provided for each marker in the Data Tabs displays includes mode (e.g., playback, record), speed, time, track number, and other such parameters that delineate how the data were recorded at a given tape location. The Data Tabs display can also be called up from the Control Panel. Note that markers are colored (e.g., during the demo they were red) but the meaning associated with the use of color here is unclear.

The recorder management table window provides a button-activated function that allows the operator to highlight all markers containing LOS data. This button is used to schedule segments of data for downlist. Data recorded during LOS is considered the most important data to downlist because it was not seen on the ground during real-time support. The next highest priority on data downlist is noisy data. Previously, this distinction was performed manually. See figure 6-5 for an example of a recorder management table.

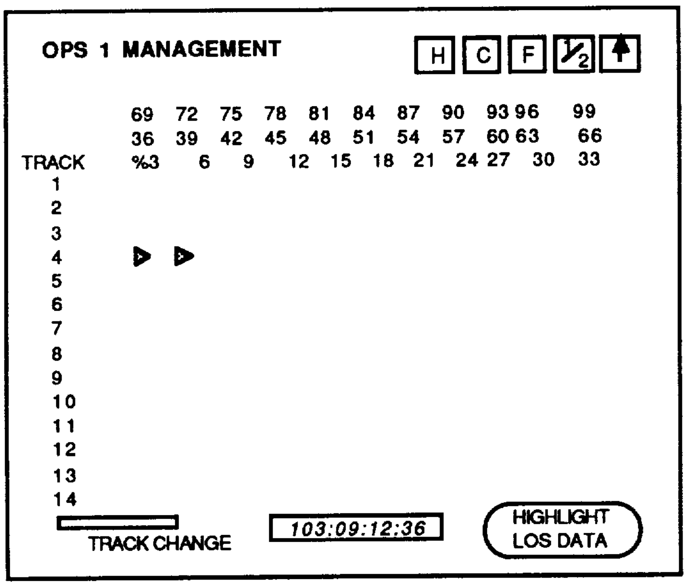


Figure 6-5. OPS 1 Recorder Management Table

The quick look summary window contains three graphical plots that provide an easily-scanned assessment of the status of the recording process. These plots contain (1) percentage of tape containing data recorded during AOS, (2) the percentage of tape containing data recorded during LOS, and (3) the head temperature with respect to time. The AOS/LOS distinction is important, since data recorded during LOS has a higher priority for retrieval than data recorded during AOS. The head temperature is a good metric for the health of the physical recording unit. See figure 6-6 for an example of the quick look summary window.

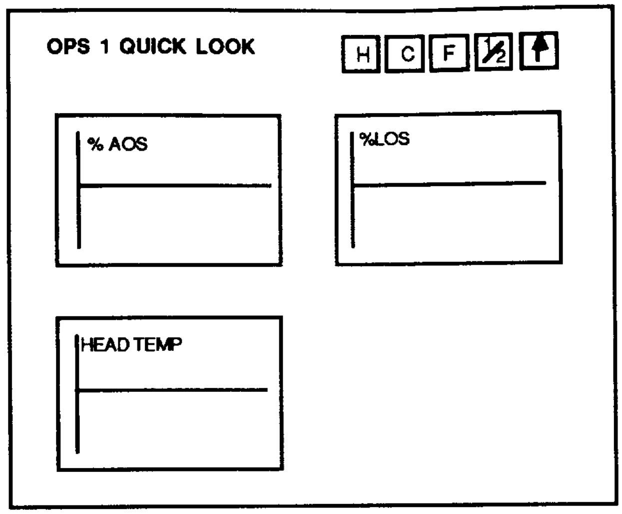


Figure 6-6. OPS 1 Quick Look Summary

- Recorder Data Logs
  Three data log icons are available, one for each data recorder. Selection of a data log icon pops up a summary of the logging activity for the specified recorder. Mode changes (i.e., off, record, playback) of the recorder are logged as events in this file. From this window, the operator can scan the contents of the selected log file and thus determine the activities of the data recorder. The characteristics displayed are the number of logs dumped, not-dumped, or lost and whether the data were recorded during AOS or LOS. A report of all events with the desired characteristics can be generated by selecting the check box to the right of those characteristics.

  The ability to search for data from a specific time region is also possible by selecting the DATA LOCATOR button. This button calls up a window requesting time region (i.e., start time and stop time). When initiated, this function searches for the status of the data recorded during the specified time region (e.g, dumped, still on recorder, etc.). This assists the DATA COMM controller in determining the status of data requests made by other flight controllers (e.g., critical data not yet provided by the Near Real Time (NRT) ground system can be tracked to see if the request has been satisfied). See figure 6-7 for an example of a recorder data report.

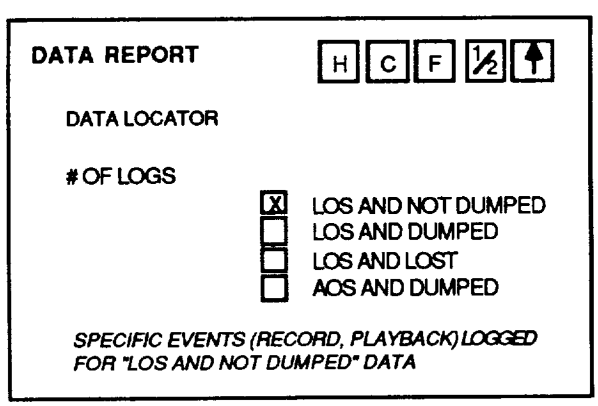


Figure 6-7. Recorder Data Log

- System Messages

  A scrollable window listing time-sorted system messages from the intelligent system is accessed via this icon. The icon appears to change colors (e.g., from white fill with red outline to solid red) at a state change, possibly when a new message is logged, although this could not be confirmed. Within the system message window, panels around each text message are also color-coded. System messages shown in the demonstration were colored red. See figure 6-8 for an example of a system message.

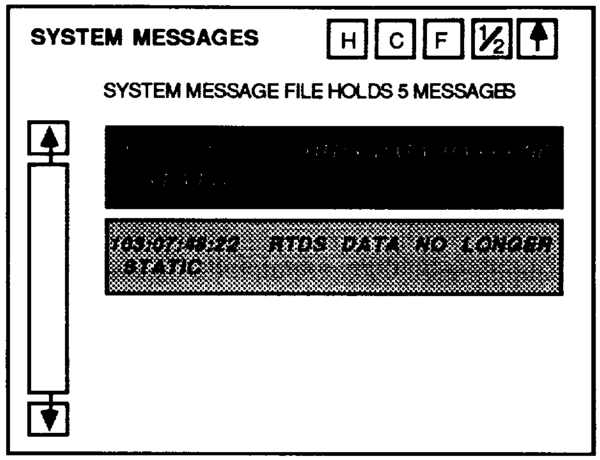


Figure 6-8. Example of System Messages Window

- Fault Messages

  A scrollable window listing time-sorted fault messages from the intelligent system is accessed via this icon. The fault message icon also appears to change colors at some state change, similar to system messages. Within the fault message window, panels around each text message are also color-coded. Fault messages shown in the demonstration were colored yellow. A possible enhancement to the fault message window is that the window would automatically popup when something off-nominal occurs. See figure 6-9 for an example of a fault message.

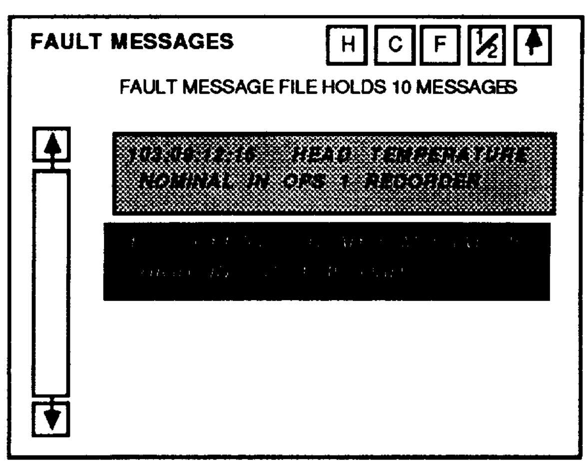


Figure 6-9. Example of Fault Messages Window

Command Entry Window

This window allows the operator to use mouse selection to enter commands for the onboard data recorders. A command window is available for each recorder. Commands include stop, playback, record, and specification of speed and track. Uplink of commands is not possible for the RTDS system, since it provides passive data acquisition only, but will be available with the Mission Control Center Upgrade (MCCU). MCCU will provide workstations and Local Area Network (LAN) data distribution for Space Shuttle flight support. Figure 6-10 shows the command entry window.

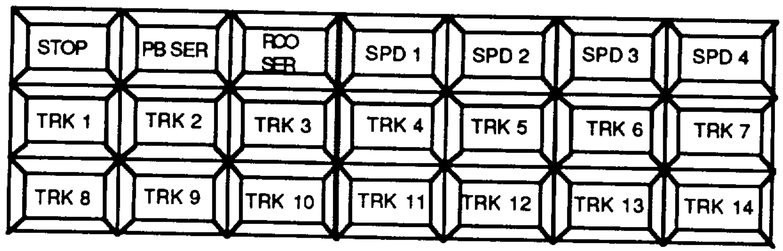


Figure 6-10. Command Entry Window

6.5 Design Process

The DATA COMM Expert System is being developed by Space Shuttle flight controllers. Thus, the user (i.e., flight controller) is also the domain expert and the software developer. Such systems are obviously characterized by early, active user involvement. The development process includes rapid, iterative prototyping with quick deployment into an operations-like environment for side-by-side testing with the existing support displays. Such side-by-side testing has proven effective in integrating the new technology into the existing support environment for other RTDS applications. At the time of the interview (July, 1990), the first phase of prototyping had just completed and the system was being prepared for testing. Data from integrated training simulations were planned for use during testing. The DATA COMM Expert System uses the RTDS data acquisition support system.

6.6 Study Method

All information about the DATA COMM Expert System was obtained by interview of the project representative and demonstration of the prototype on July 9, 1990. The project representative George Pohle is both a flight controller and a system developer.

Study Team

- Debra Schreckenghost (The MITRE Corporation)

Project Representative

- George Pohle (Rockwell Shuttle Operations Company)

6.7 Case Data Sources

No written information was available for the DATA COMM Expert System. Since hardcopies of screens were not available, drawings used throughout this section are based on observations made during the demonstration.

Section 7
Payload Deployment and Retrieval System (PDRS) Decision Support System (DESSY)

7.1 System Description

DESSY is an intelligent system being developed for Space Shuttle flight operations support of the PDRS ground-based flight console position during both missions and training simulations. DESSY will support the flight controllers in most of the tasks that they perform. The PDRS flight controllers are responsible for the Space Shuttle Remote Manipulator System (RMS), the teleoperated robotic arm used to move objects into and out of the payload bay of the vehicle. The primary objectives of the PDRS flight controllers are (1) to assess total RMS performance, status, and configuration for the Flight Director and crew, and (2) to correct anomalies in RMS operations or configuration where possible. To achieve these objectives, a number of tasks must be performed by a PDRS operator. These tasks may be subdivided into three main groups: assessment of the current situation, diagnoses of faults, and recommendation of corrective procedures. Specific tasks are listed in table 7-1.

Table 7-1. Tasks Performed by the PDRS Flight Controllers (JSC, 1983)

Tasks Performed by the PDRS Flight Controllers

- Monitor and verify the following information
  - Parameter values of PDRS system)
  - Status and configuration of related, peripheral systems
  - Selection and execution of procedures and crew activities
- Advise ground control and crew of system status and configuration
- Provide situation assessment of operations capability (i.e., how anomaly affects crew safety and mission goals, remaining operational capability after an anomaly?)
- Recommend methods for diagnosis of faults and procedures for corrective action
- Assist in performing diagnosis of faults
- Assist crew with corrective action
  - Monitor procedure execution
  - Generate and test new corrective procedures assisted by off-line simulation
- Alert the occurrence of Caution and Warning alarms
- Event detection and recording (i.e., logging)
- Evaluate and perform test and checkout of
  - RMS trajectories and maneuvers
  - Contingency corrective procedures
- Coordinate command loads to uplinked to the RMS
- Evaluate arm performance and performance trends
- Enforce flight rules relative to RMS operations
- Monitor and support RMS Detailed Test Objectives and coordinate real-time assessment of test sequence

The PDRS consists of all elements of the vehicle used during operation of the RMS. These elements include:

- RMS, a mechanical appendage attached along the port longeron of the vehicle with three joints, a 2 Degree of Freedom (DOF) shoulder, a 1 DOF elbow, and a 3 DOF wrist.
- Payload Retention Latch Assemblies (PRLAs) that secure payloads in the payload bay

- Manipulator Positioning Mechanisms (MPM) and Manipulator Retention Latches (MRL) that attach the RMS to the vehicle

- Payload handling aides and guides

- Closed Circuit TeleVision (CCTV) system that provides visual guidance to the crew during RMS operations

- General Purpose Computers (GPC) that monitors status and provides software control for vehicle systems, including RMS

The RMS provides the capability to manipulate a payload out of or into the Space Shuttle payload bay. Since the RMS interfaces directly with payloads, anomalies can have an immediate impact on both vehicle and payload operations.

PDRS ground-based flight control support consists of four personnel, the RMS Officer (located in the Flight Control Room, or FCR) and three RMS support personnel (located in a Multi-Purpose Support Room, or MPSR). The responsibilities of each of these flight control positions are:

- RMS Officer
  Coordinates the ground-based RMS support and provides status and configuration to the Flight Director and other console positions; located in

- RMS Systems
  Monitors the RMS hardware systems

- RMS Software
  Monitors the RMS software systems

- RMS Support
  Provides assistance is assessing RMS performance, checking RMS command loads uplinked to the GPCs, evaluating RMS trajectories and maneuvers, and generating contingency RMS procedures

The PDRS DESSY is a rule-based system originally developed using CLIPS on a PC platform. A verification and validation tool Rule Checker, developed by the the Intelligent Systems Branch (ISB) at the Johnson Space Center, was used to check the initial rule base for consistency and completeness. After preliminary testing, DESSY was ported to the G2® environment on a VAX® 3100 workstation in the Real Time Data System (RTDS) prototyping lab. This port was an attempt to comply more closely with the constraints of the delivery environment (e.g., UNIX™ workstation, the X Window System™, C) and to allow use of the RTDS as a telemetry data source. Ultimately, the delivery environment for this application will be the X Windows System and a rule-based intelligent system tool (possibly G2) on a Masscomp workstation.

® - G2 is a registered trademark of Gensym Corporation.
® - VAX is a registered trademark of Digital Equipment Corporation.
™ - UNIX is a trademark of AT&T Bell Laboratories.
™ - X Window System is a trademark of MIT.

Two pieces of PDRS flight support software were previously developed for use on a Masscomp workstation in the FCR and are being used in parallel with the existing flight support systems: the Position Monitor and the Operations Monitor. DESSY will be used in conjunction with these applications. The Position Monitor provides a graphical representation of the three-dimensional projection of the physical configuration of orbiter, RMS, and payload driven by downlisted telemetry. Formerly, the only available graphical display of the RMS was generated by off-line simulation capability (i.e., stand-alone systems not connected to real-time data). The Operations Monitor performs limit sensing on selected telemetry parameters to detect out-of-range values. The PDRS DESSY, currently resident in the RTDS prototyping lab, will eventually be available on workstation in the FCR as well.

7.2 Intelligent System and Functions

At the time of this report, the DESSY rule-based prototype only supports a portion of RMS operations, specifically deploying and stowing the RMS. It consists of rule bases for the MPM and MRL subsystems. The purpose of this prototype is to assess the current state of these subsystems, detect anomalies associated with a subsystem, and to assist is identifying the cause of detected anomalies. The PDRS includes four MPMs, one located at the shoulder of the RMS and three others located along the arm (i.e., fore, mid, and aft attachment points), and three MRLs, co-located with the fore, mid, and aft MPMs. The MPMs are rotating mechanisms to move the RMS away from the Space Shuttle body prior to deployment and toward the Space Shuttle body prior to stowing. The MRLs are latches located near the MPMs that attach the RMS to the Space Shuttle body. Figure 7-1 defines the components of the MPM/MRL subsystem on an overview of the RMS.

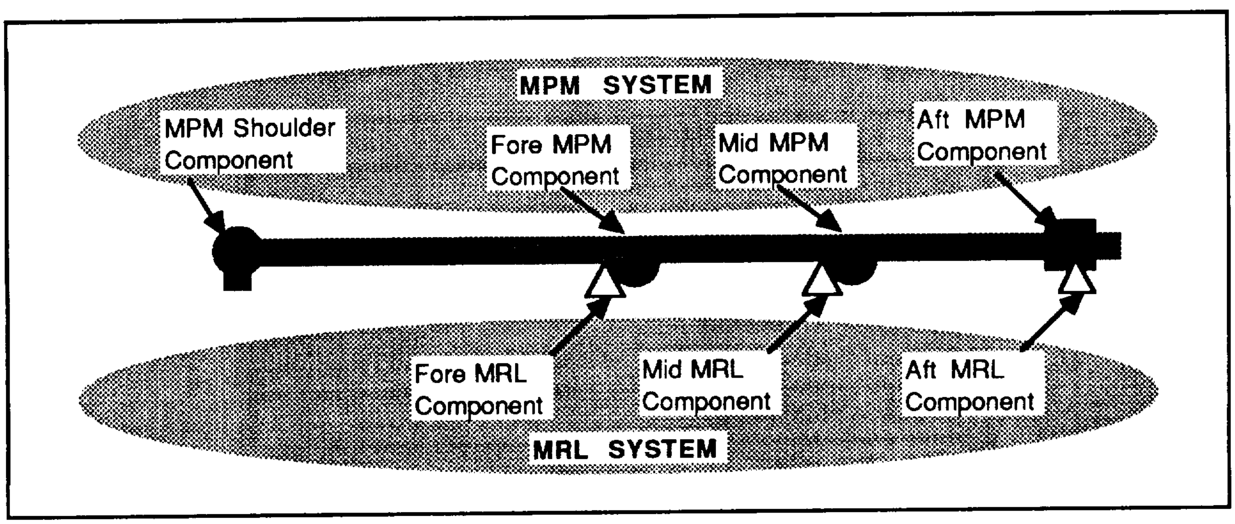


Figure 7-1. Definition of Components of the MPM/MRL Subsystem

The MPM portion of the rule base was developed first. The information represented in this rule base is used for two functions: detecting anomalies and identifying the cause of those anomalies. Domain knowledge is represented at the subsystem (i.e., MPM) level only and is not separated into component (i.e., shoulder, fore, mid, and aft) level, although physically there are four MPMs and the downlist contains distinct parameters for each of these locations.

The information currently used by the intelligent system for anomaly detection can be classified into three areas:

- Subsystem state
  The current condition of the subsystem relative to a pre-defined set of possible conditions (e.g., MPM stowed or deployed)

- Subsystem status
  An evaluation of the ability of a subsystem to support nominal operations (e.g., MPM nominal, single motor capability, etc.) based on an assessment of subsystem behavior (nominal or anomalous)

- Subsystem command
  A crew request to initiate action in the monitored subsystem or to alter the information used by that subsystem (e.g., crew commands deployment of MPM by activating a switch on the Display and Control panel)

The information currently used by the intelligent system for isolating the cause of an anomaly are:

- Fault symptoms
  Data parameters or other information, such as information heard on the voice loop, whose current values indicate the presence of a fault

- Fault ambiguity groups
  Set of faults indistinguishable using the available fault symptoms

Figure 7-2 summarizes the information represented in the MPM rule base.

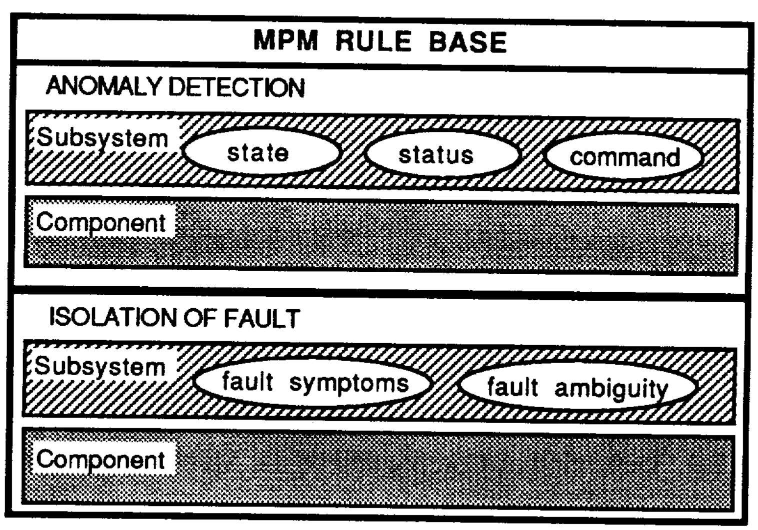


Figure 7-2. Information Within the MPM Rule Base

The MRL rule base was developed after the MPM rule base. In this rule base, state, status, and command are also the key types of information assessed. The representation differs from the MPM knowledge representation, however, since both a component level and subsystem level assessment is performed. Thus, the MRL rule base allows detection of anomalies at both the subsystem and component levels. Fault symptoms are also identified and associated with specific faults. Fault ambiguity information has not yet been included in this rule base. Figure 7-3 summarizes the information represented in the MRL rule base.

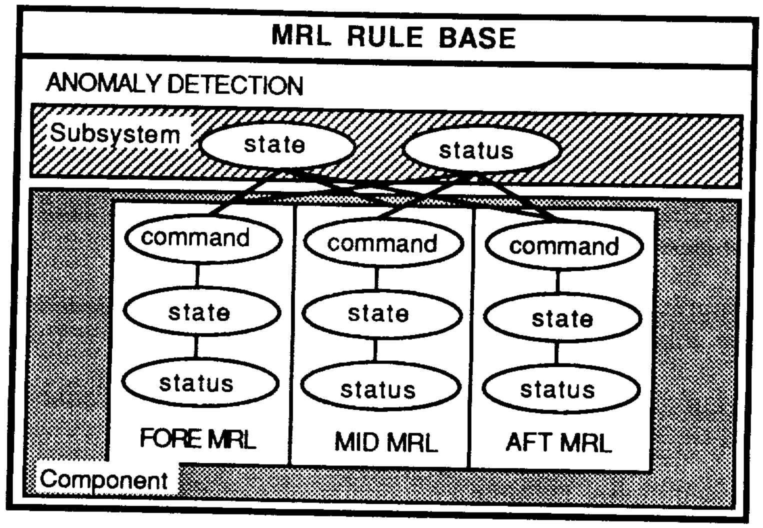


Figure 7-3. Information Within the MRL Rule Base

7.3 Human-Intelligent System Interaction Functions

DESSY is designed as a passive monitor of the PDRS subsystems. It can recommend the execution of actions, but cannot initiate those actions. For the existing prototype, the flight controller has limited intervention and control capability, consisting of the input of information not available on the telemetry downlist (e.g., information heard over the voice loop) and stopping and restarting the intelligent system. The HCI design concepts outline requirements for additional intelligent system intervention and control capabilities. The HCI design concepts identify the following required capabilities for operator's using the intelligent system:

- Create a checkpoint and restart system from checkpoint
- Pause, stop, and exit the intelligent system
- Log and review of intelligent system information
- Playback using recorded telemetry data
- Control the archiving of telemetry data; telemetry acquisition is controlled from a separate workstation
- Enter data unavailable on downlist
- Correct erroneous system information, including alteration of parameters internal to the intelligent system
- Pre-process telemetry or disable its use in intelligent system

- Direct the inferencer to operate on a particular rule set first
- Selectively enable and disable portions of the knowledge base (i.e., rule sets, such as MPM or MRL rule sets)
- Select alternate screen formats for display
- Select alternate windows containing different information for display within the dynamic workspace of a screen

The HCI design concepts also identify the types of information that should be available to the operator. The major types of information include:

- Time and mission configuration (i.e., mission specific identifiers, such as vehicle identifier, mission identifier, current software load in onboard GPCs)
- Mission context (i.e., an indicator of position within the planned mission activity timeline, such as a sequence of PDRS operational phases shown in figure 7-4)
- User control inputs, including parameters not on the downlist and console support information (e.g., operator identifier at a shift change, or handover)
- Subsystem state, status, and commands
- State of the microswitches (i.e., electrical switches in the MPM/MRL subsystem responsible for command, display, etc.) associated with the MPM/MRL subsystem
- Event messages, including timetag, source of message, and content of message
- Recommendations
- Schematics and tables of design information
- Parameter history for state, status, crew commands, and microswitches
- Potential faults in fault ambiguity group, subsystems affected by potential faults, procedures or activities to resolve ambiguity, and justification for procedures (flight rules, references)
- Log files containing intelligent system output and telemetry data
- Checkpoint file containing all system data at a given time
- Configuration of the intelligent system (e.g, rule bases currently active, telemetry parameters disabled or filtered), including time of alteration, type of alteration, and operator performing alteration
- Operator-input data values (i.e., parameters not available on the downlist)
- Identifier of the current operator

Sources of this information include telemetry downlist, COMPS, the intelligent system, and the flight controller.

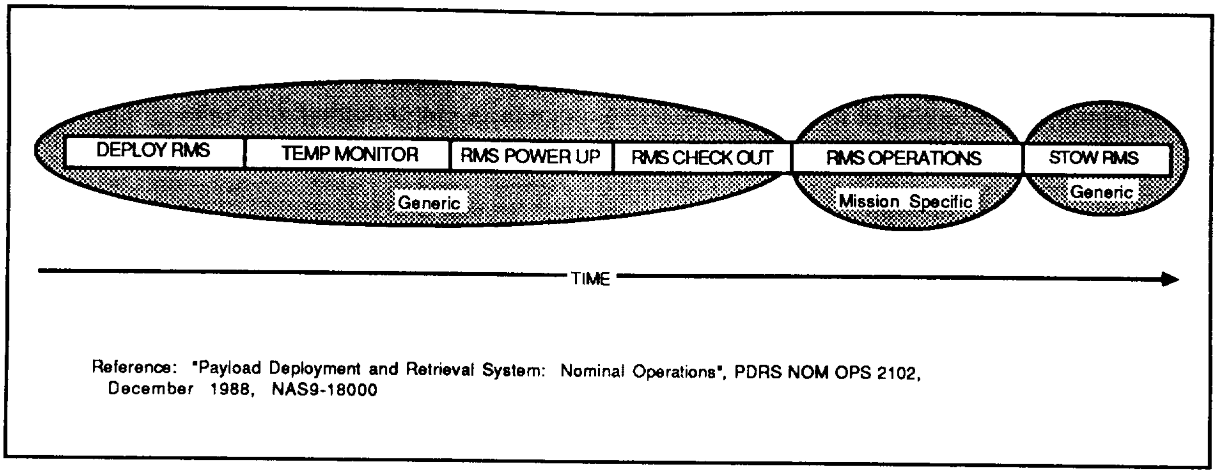


Figure 7-4. Example of Mission Context: Phases of PDRS Operations Support

There are three modes of operation described in the HCI design concepts: (1) real-time monitoring, (2) review of intelligent system results, and (3) playback using recorded data. In real-time monitoring mode, the intelligent system is processing and displaying information in real time. In review mode, the operator can use display formats available in real-time to review recorded output from the intelligent system. In playback mode, the intelligent system re-executes using recorded input data. Review capability provides a limited form of collaboration between the operator and the intelligent system. Playback capability assists the operator in intervening in intelligent system processing.

The PDRS HCI design concepts provide a variety of capabilities for the operator to intervene into intelligent system processing. One type of intervention is manipulation of input to the intelligent system. The operator can disable parameters input to the intelligent system and can provide information to the intelligent system not available on the downlist. He can filter telemetry parameters prior to use in the intelligent system. He can also restart the intelligent system from checkpoint (i.e., intelligent system state information saved at some time in the past).

The HCI design concepts specify access to information useful in monitoring intelligent system performance (e.g., Central Processing Unit (CPU) and memory usage). They also provide some capability for improving performance in real time (e.g., ability to disable rule sets).

7.4 Supporting User Interface Capabilities

The workspace for DESSY is planned to include the following full screen displays:

- Integrated Status
- MPM/MRL Subsystem Status
- Arm-Based Electronics (ABE) Subsystem Status
- Display and Control (D&C) Subsystem Status
- End Effector (EE) Subsystem Status
- Manipulator Controller Interface Unit (MCIU) Subsystem Status
- COMPS (i.e., computed data based on raw telemetry from the ground-based Mission Operations Computer)
- Telemetry

Specific requirements have been developed for the Integrated Status Display and the MPM/MRL Subsystem Status Display. All other displays remain to be defined. Figure 7-5 summarizes the hierarchy of available displays expected for the complete DESSY.

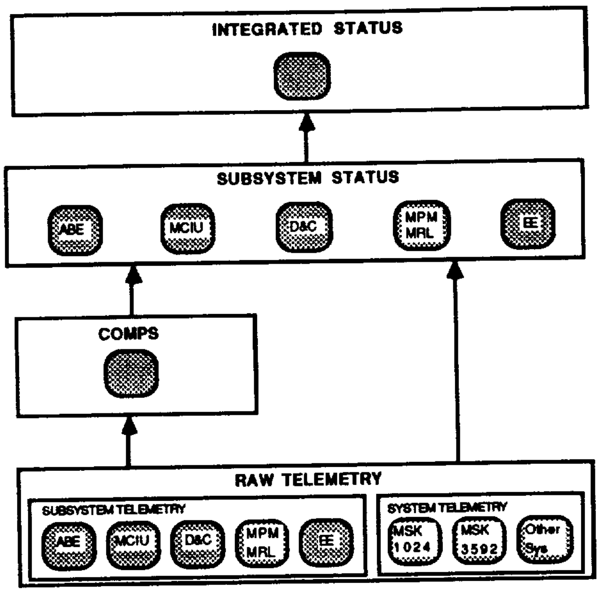


Figure 7-5. Screen Definitions for PDRS Information Layers

The initial MPM/MRL prototype had a textual user interface designed for use during development only. Requirements for an operational user interface were needed. Human-computer interaction (HCI) design concepts were developed for the PDRS DESSY to assist in the development of these requirements. The design concepts addressed two portions of the user interface: a top-level display for assessment of integrated status of all PDRS subsystems and a display of information specific to the MPM and MRL subsystems. The discussion of the user interface for the PDRS DESSY contained in this report is based on the HCI design concepts. These design concepts are currently being used to develop a user interface for DESSY.

It was recognized that the user interface should permit assessment of an integrated status for all PDRS subsystems. The user interface design includes an Integrated Status Display that would be viewed by the flight controller during nominal operations. Other displays would be available for access to detailed subsystem information. These displays would be accessed as needed by the flight controller to support specific mission activities, such as anomaly diagnosis and malfunction correction.

Both the Integrated Status Display and the MPM/MRL Subsystem Status Display present important events as messages in scrollable message list. The source of each message is included in the message field. For the Integrated Status Display, events can be alternately displayed on a timeline with events grouped by subsystem.

Consistent color-coding is used throughout the design concepts and reflects coding often used in other RTDS systems:

- Green: nominal
- Yellow: cautionary
- Red : failure
- Orange: data unavailable due to Loss of Signal (i.e., static data)
- Blue : background color used to associate related items that are not physically co-located

Integrated Status Display

The Integrated Status Display is designed to quickly orient the operator about the current behavior of the PDRS subsystems. The workspace of this display contains the following major display regions:

- Time and Mission Configuration
- Operator Control Buttons
- Phase of RMS Operations
- Subsystem Status and State
- Event Summary Messages about all RMS subsystems
- Recommended Activities concerning all RMS subsystems
- Dynamic Workspace

All regions of this display contain a fixed display format, except the Dynamic Workspace. The contents of this region are selected during real-time operations by the flight controller. The display options currently identified for this region include (1) summary of important events displayed on timeline, (2) schematics, such as an overview major RMS subsystems. See figure 7-6 for an example of the Integrated Status Display.

The HCI design concepts use three-dimensional icon buttons similar to those provided by the PDRS Position Monitor as a means of accessing operator control options. Capabilities for interacting with the intelligent system from the Integrated Status display include:

- Screen select

  Select a screen for display from the hierarchy of available screens (figure 7-5)

- Intelligent system control
  1) Start, stop, pause, and reset
  2) Selectively focus on a rule set or disable rule sets; rule sets are partitioned by hardware subsystem
  3) Time-synchronize intelligent system with data source
  4) Monitor execution of intelligent system through CPU and memory usage

- Review of logged information

  Visual inspection of information recorded during execution of intelligent system
  1) View logged information using same display formats as available real-time
     a) Use a different screen background to distinguish from real-time
     b) Step through review for close inspection
  2) View operator changes to baseline intelligent system configuration (e.g., at operator handover)

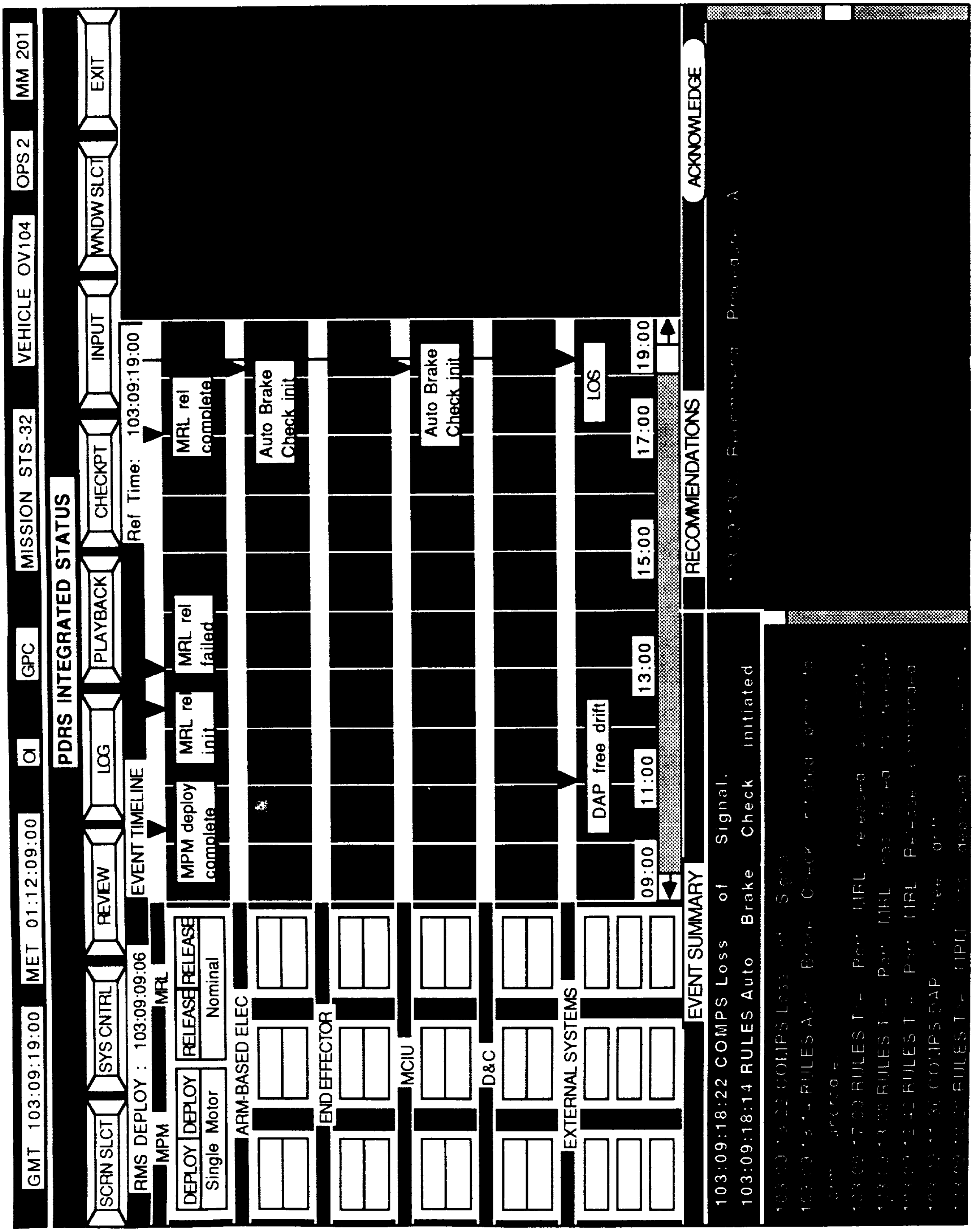


Figure 7-6. Example of the Integrated Status Display

ORIGINAL PAGE IS
OF POOR QUALITY

- Log of run-time information
  Record selected information generated during execution of intelligent system, such as
  1) Subsystem state, status, and crew commands
  2) Events and messages
  3) COMPS and telemetry
  4) User data inputs

- Playback of telemetry data
  Execute the intelligent system using recorded telemetry data
  1) Operator controls include
     a) Specify a telemetry data file
     b) Start, stop, pause
     c) Execute stepwise, 1 time sample of data per execution step
     d) Move forward and backward in time through the telemetry data file
  2) Results of playback can be reviewed using any display format available in real-time

- Checkpoint
  Save all intelligent system information at a given time to allow restart of system from that time
  1) Create a checkpoint
     a) Immediate: save current system state
     b) Periodic: periodically save system state
     c) Event-driven: save system state when an important event occurs (e.g., Loss of Signal, change of RMS phase)
  2) Load a checkpoint saved previously

- User data inputs
  Specify information needed by intelligent system but unavailable or incorrect on downlist
  1) Enter unavailable information
     a) Parameters not on telemetry downlist (e.g., voice data)
     b) Console support information (e.g., handover)
  2) Correct for erroneous system information
     a) Create data filters and value limits
     b) Enable and disable use of data

- Window select for dynamic workspace
  Select window for display in dynamic workspace
  1) Event Timeline: important events grouped by subsystem and displayed on a timeline
  2) RMS Overview Schematic: status displayed on a schematic that illustrates all components of the PDRS, including peripheral systems
  3) RMS Heaters Schematic

- Exit intelligent system
  Graceful shutdown of the intelligent system (e.g., before exiting, close log files or finish current data cycle); includes the option to take a checkpoint prior to exit

The HCI design concepts provide access to fault ambiguity information by mouse-selection of a fault ambiguity message from the event summary message list. The fault ambiguity popup window provides the following information:

- Systems containing potential faults
- Potential fault for each system
- Suggested procedures or activities to resolve ambiguity
- Justification for suggested procedures (e.g., flight rules, references)

A possible enhancement of this capability is an operator-initiated WHAT-IF evaluation of the mission impact of executing the suggested procedures.

MPM/MRL Subsystem Status Display

The MPM/MRL Subsystem Status display is designed to provide the operator with access to detailed information on the MPM and MRL subsystems. The workspace of this display contains the following major display regions:

- Time and Mission Configuration
- MPM and MRL Status
- Operator Control Buttons
- Event Summary Messages about all RMS subsystems
- Detailed MPM/MRL Messages
- Dynamic Workspace

Similar to the Integrated Status display, all regions of the MPM/MRL Subsystem Status Display contain fixed display formats, except the Dynamic Workspace. The operator-selectable display options currently identified for this region include (1) history of information from telemetry downlist and intelligent system, (2) detailed information about the microswitches, (3) tabulated design information, (4) subsystem schematics. See figure 7-7 for an example of the MPM/MRL Subsystem Status display.

Detailed MPM/MRL Messages provide the equivalent of a rule trace of intelligent system reasoning. Used in conjunction with a summary of the configuration changes to intelligent system, this rule trace provides insight about how the intelligent system reached its conclusions.

Operator control buttons are also provided by the MPM/MRL Subsystem Status Display. These buttons include three control options similar to those provided by the Integrated Status Display (i.e., screen select, user data inputs, and exit) and two new control options:

- Tables
  Select tabulated design or operations information for MPM/MRL subsystem
- Schematics
  Select schematic design or operations information for MPM/MRL subsystem

The MPM/MRL Subsystem Status Display has mouse-sensitive regions associated with the display of MPM/MRL states, MPM/MRL status, MPM/MRL commands, and microswitch states. Selection of these sensitive regions allows access to additional information about the associated parameter. For MPM/MRL state, status, and command, a message list with timetags indicating previous values of the parameter is available. For microswitch state, the operator can display the circuit containing the microswitch, a telemetry display item describing the downlisted parameter representing the state of the microswitch, and a plot of the previous values of the microswitch (see figure 7-7 for an illustration).

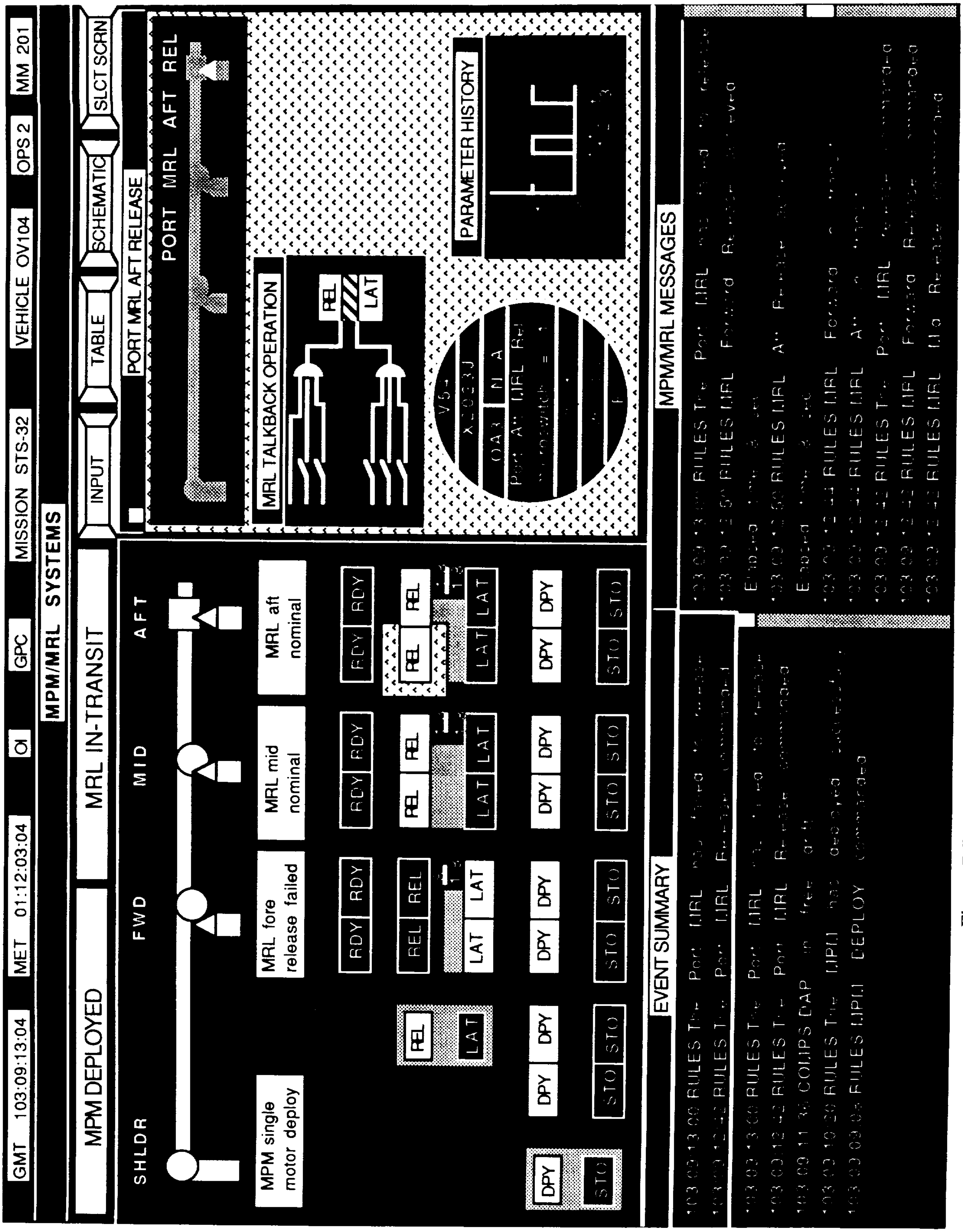


Figure 7-7. Example of the MPM/MRL Subsystem Status Display

7.5 Design Process

There have been a number of participants in the design and development of the PDRS DESSY. This prototyping effort was conceived and implemented by PDRS flight controllers, in particular Don Culp from the Rockwell Shuttle Operations Company (RSOC), Joe Watters (formerly RSOC), and Kristen Farry (formerly RSOC). Other PDRS flight controllers have been periodically consulted for review of requirements and designs. The PDRS flight control group that is developing DESSY has an informal but active working relationship with Dr. Jane Malin of the Intelligent Systems Branch (ISB) of the Engineering Directorate (ED) at Johnson Space Center (JSC). Dr. Malin, with both government employee and contractor support from Lockheed Engineering Support Company (LESC) and the MITRE Corporation, has provided consultancy during development of the prototype. Consultancy services have included knowledge engineering expertise, CLIPS expertise, verification and validation assistance, and HCI requirements development.

The PDRS DESSY design approach separates the prototype development into five distinct sections, corresponding to the five major hardware subsystems of the RMS:

- Manipulator Positioning Mechanisms (MPMs) & Manipulator Retention Latches (MRLs)
- Arm-Based Electronics (ABE)
- Display & Control (D&C)
- End Effector (EE)
- Manipulator Controller Interface Unit (MCIU)

The MPM/MRL subsystem was selected for the initial prototype, since the knowledge base was perceived as easily separated from the other subsystems (i.e., well-defined, non-mission specific procedures that do not rely heavily on other subsystems). This prototype was developed by PDRS flight controller Don Culp with knowledge engineering support and CLIPS consultancy from contractors supporting the ISB (Culp, 1990). DESSY has been developed using an iterative prototyping technique with active user involvement. This prototype is planned for testing in an operations-like environment during a Space Shuttle mission in first quarter of fiscal year 1992.

The case study team was involved in the design of DESSY. They developed the PDRS HCI design concepts to assist in defining requirements for DESSY. These design concepts were based on documents used by flight controllers during training and mission support, working meetings with flight controllers, and earlier HCI design work done by Gordon Johns from the MITRE Corporation. The HCI design concepts were developed as a paper storyboard using MacDraw™ on a Macintosh™ SE. The HCI design concepts were modified based on review by personnel with expertise in rule-based systems, human factors, software prototyping, and PDRS flight control.

The storyboard of design concepts were implemented as a user interface prototype by the HCI Lab in the Man-Systems Division of the Space and Life Sciences Directorate at JSC on a Macintosh 2 Workstation using a HCI prototyping tool called Prototyper. This electronic version of the design concepts has been used to demonstrate existing and planned capability. Another planned use of the prototype is hand-ons testing and evaluation of the design by PDRS flight controllers. During the translation of the PDRS design concepts from a storyboard format to an electronic prototype, it was observed that a number of design compromises and

---

™ - MacDraw is a trademark of Apple Computer, Inc.

™ - Macintosh is a trademark of Apple Computer, Inc.

constraints were imposed by the prototyping tool. Since this prototyping tool will not be used for the operational application, these constraints do not represent actual design constraints. The design is still undergoing significant modification. In the interim, portions of the MPM/MRL HCI design are being implemented and interfaced to the existing prototype as a way of testing their feasibility and utility.

7.6 Study Method

Information about the PDRS DESSY was obtained by interview of users and developers, direct participation in the development of HCI design concepts, and review of the case data sources cited below. These activities were on-going throughout fiscal year 1991.

Study Team

- Jane Malin (NASA Johnson Space Center)
- Debra Schreckenghost (The MITRE Corporation)

Project Representatives

Domain expertise and prototyping
- Don Culp (Rockwell Shuttle Operations Company)
- Joe Watters (formerly Rockwell Shuttle Operations Company)
- Kristen Farry (formerly Rockwell Shuttle Operations Company)

Real Time Data System software implementation
- Mark Gnabasik (The MITRE Corporation)

Human-computer interaction and human factors
- Debra Schreckenghost (The MITRE Corporation)
- Jane Malin (NASA Johnson Space Center)
- Mary Czerwinski (formerly Lockheed Engineering and Science Company)
- Benjamin Beberness (Lockheed Engineering and Science Company)
- Gordon Johns (The MITRE Corporation)

Knowledge engineering and CLIPS expertise
- Dale Phinney (Lockheed Engineering and Science Company)

Verification
- Jane Malin (NASA Johnson Space Center)
- Jodi Seaborn (NASA Johnson Space Center)

7.7 Case Data Sources

Culp, Donald R. (August, 1990), "Interpretation of Space Shuttle Telemetry", *Proceedings of First CLIPS Conference*, Johnson Space Center, Houston, TX: NASA.

Culp, Donald R. (1990), listing of rule base for Manipulator Positioning Mechanisms and Manipulator Retention Latches, Houston, TX: Rockwell Shuttle Operations Co.

Farry, Kristen (August, 1987), draft data flow charts for Space Station fault management, Houston, TX: Rockwell Shuttle Operations Co.

Hughes, J.M. (January, 1989), *Real Time Data System Remote Manipulator System Position Monitor Level A Requirements*, Houston, TX: Rockwell Shuttle Operations Co.

Johns, Gordon (May, 1989), "First-Cut Interface for MPM/MRL", draft, Houston, TX: The MITRE Corporation.

Johns, Gordon (November, 1989), "Requirements for the Joint Motor Display, a Component of the PDRS RTDS System Interface", draft, Houston, TX: The MITRE Corporation.

JSC (October, 1983), *PDRS Console Handbook*, JSC-17525, Volume 1, October 14, 1983.

JSC (February, 1988), *Payload Deployment and Retrieval System: Overview Workbook*, PDRS OV 2102, NAS9-18000, TD383.

JSC (April, 1988), *PDRS Malfunction Workbook*, PDRS MAL 2102, NAS9-18000, TD385.

JSC (December, 1988), *Payload Deployment and Retrieval System: Nominal Operations*, PDRS NOM OPS 2102, NAS9-18000, TD386.

JSC (January, 1989), *Remote Manipulator System Failure Modes and Effects Analysis and Critical Items List*, JSC-22373.

JSC (December, 1989), *PDRS Console Handbook, Volume II*, JSC-17525, Johnson Space Center, Houston, TX: NASA.

Kerr, Ronald L. (February, 1989), *Level A Requirements for RTDS Temperature Mode Monitor*, Mission Operations Directorate, Johnson Space Center, Houston, TX: NASA.

Kerr, Ronald L. (March, 1989), *Level C Requirements for RTDS Temperature Mode Monitor*, Mission Operations Directorate, Johnson Space Center, Houston, TX: NASA.

Seaborn, Jodi (July, 1989), "Phase I Technical Intern Review", briefing charts, Intelligent System Branch, Engineering Directorate, Johnson Space Center, Houston, TX: NASA.

Schreckenghost, Debra (July, 1990), "PDRS Intelligent System Human-Computer Interface Design Concepts", briefing charts, MITRE, Houston, TX.

Section 8
Remotely Augmented Vehicle Expert Systems (RAVES)

8.1 System Description

RAVES is planned for use in the NASA Integrated Test Facility (ITF), which is supports integration and test of advanced integrated aircraft, such as the X-29. Tests of both ground and flight operations are possible. The facility can be configured for both real-time testing with the vehicle in the loop and standalone testing for preflight and postflight support.

RAVES is currently located in the Remotely Augmented Vehicle (RAV) Lab of Ames Research Center's Dryden Flight Research Facility. The RAV can control an aircraft from the ground. The primary purpose of RAVES is to support real-time monitoring of control parameters during flight tests of the X-29. As such, it is considered to be flight critical software. RAVES was designed to allow experimentation with different control algorithms for dynamic, closed loop control in very time-constrained situations (on the order of 2.5 millisecond delays). The application monitors data from the aircraft telemetry stream and from ground-based systems that perform control calculations.

RAVES was built using the Real Time Data System (RTDS) hardware and data acquisition software, with the addition of the VI DataViews® software for graphical interfaces. The RTDS portion of this application monitors the health of the ground-based systems and the telemetry stream to verify command uplinks and detect errors on the downlist. The expert system looks at groups of individual sensed failures as indicators of higher level faults (identifying the probable cause of a fault by monitoring clusters of sensor signatures). The rules (i.e., if-then constructs) are written in C with an interface in Data Views, instead of the Masscomp graphics package delivered with the RTDS software. Data Views was used to allow users to easily develop their own interfaces. The default sample rate has also been changed from 1 sample per second used for Space Shuttle to 4-6 samples per second.

RAVES has been delivered to the RAV Lab as an operational tool for use with a variety of research vehicles. The system has been formally released for the X-29. Although the X-29 is the only vehicle used to test the system to date, other research vehicles will make use of the system in the future. The F-18 has been added to the types of vehicles to be supported by RAVES.

8.2 Intelligent System and Functions

RAVES is a rule-based system that assists ground operators in controlling the X-29 by detecting possible aircraft control problems and providing recommendations for correcting these problems. State, status and configuration of the control system is assessed. RAVES also provides information about the status of the data transmission paths in RAV system. Problems that are identified include equipment failure, software problems, and abnormal signal strength. Fault messages are generated for both types of problems. Logs of both fault messages and flight data can be made.

® - Data Views is a registered trademark of VI Corp.

8.3 Human-Intelligent System Interaction Functions

RAVES is a rule-based system that monitors control parameters from an X-29 to detect problems and provides recommendations about correcting those problems. A schematic of the onboard circuitry receiving RAV commands is used to identify aircraft faults. Tabulated control parameters illustrate state, configuration, and status of aircraft control. For tabulated displays, text is highlighted when anomalies are detected. The status of components in the data transmission paths of the RAV system are illustrated using a block diagram of these transmission paths.

RAVES provides the operator with many alternate screen formats. Screen format is selected based on the current activity of the operator. Screen formats that are not driven by the intelligent system remain active when the intelligent system is disabled, allowing the operator to continue operations without the intelligent system.

RAVE supports collaboration by providing an explanation of anomalies and specifying recommended operator actions. Faults are identified in message lists. Fault messages include coding that indicates the severity of the message. This information assists the controller in determining if the on-going activity should be interrupted to handle the incoming message.

Although control commands are not issued using RAVES, it assists in ground controllers in determining what those commands should be.

RAVES has two modes of operation:

- Real-time flight
  Downlisted telemetry is provided to the expert system from the ADS-100 during real-time or from recorded tapes.
- Playback
  Telemetry data recorded in a data log file is provided to the expert system.

Logging options depend upon the mode of operation. Two logs can be created:

- Data log
  Formatted flight data for use in playbacks and post flight analysis
- Fault log
  Fault messages from both the support system (i.e., data acquisition and operating environment messages) and the application (i.e., faults associated with the values of downlisted variables)

For real-time, data logs are created by default unless the operator disables logging. For playback, data logs are retrieved for replay and the fault log is displayed via the Fault Message Monitor window. Note that the fault log can be different from the log associated with the selected data file. Fault messages are issued in both modes.

RAVES illustrates a technique for on-the-job training. Alternate display formats are provided for novices and experts. The layout of information is the same for both formats, however. Novices are trained using a format displaying items with full English descriptors. They can compare these descriptors to the commonly used acronyms arranged in the same layout on the display format used by experts.

8.4 Supporting User Interface Capabilities

A number of displays have been made available for use during flight support. All RAVES displays are partitioned into three main sections, which resemble the layout used for many of the RTDS displays:

- Status Line containing flight and time information, located at the top of each display
- Currently selected Main Display, located in the central section of each display
- Fault Message Window containing time-sorted system and application fault messages, located at the bottom of each display

These three windows are always updated when RAVES is executing.

The Main Display portion of the screen can display any of a variety of user-selected windows. These windows are accessible by selecting the page title from the top-level RAVES display (see figure 8-1). Once a Main Display has been selected, the operator may move between alternate views directly without returning to the main RAVES menu by selecting from a set of buttons labeled with window titles. Main Display windows available from the RAVES main menu are:

1) Logging

- Logging Control
  Available during real-time mode only, this display window provides user control buttons for the logging function. Control options include:
  - Disable/enable logging
  - Change the size of the log
  - Delete a log
  - Save a log to a user specified file

  For the save option, the user selects a number between 1 and 10 that is used to identify the log. Note that each log is actually saved to four files (data log, fault log, tag-MSID, time log). See figure 8-2 for an example of the Logging Control window.

- Fault Message Monitor
  Available during playback mode only, this display window provides the user with a summary list of all fault messages received from the system and the application. See figure 8-3 for an illustration of the Fault Message Monitor

2) Format Independent Storage Array (FISA) Display
   Masscomp graphics display used to monitor and control data acquisition.

3) Ring Buffer
   Masscomp graphics display used to monitor and control data acquisition.

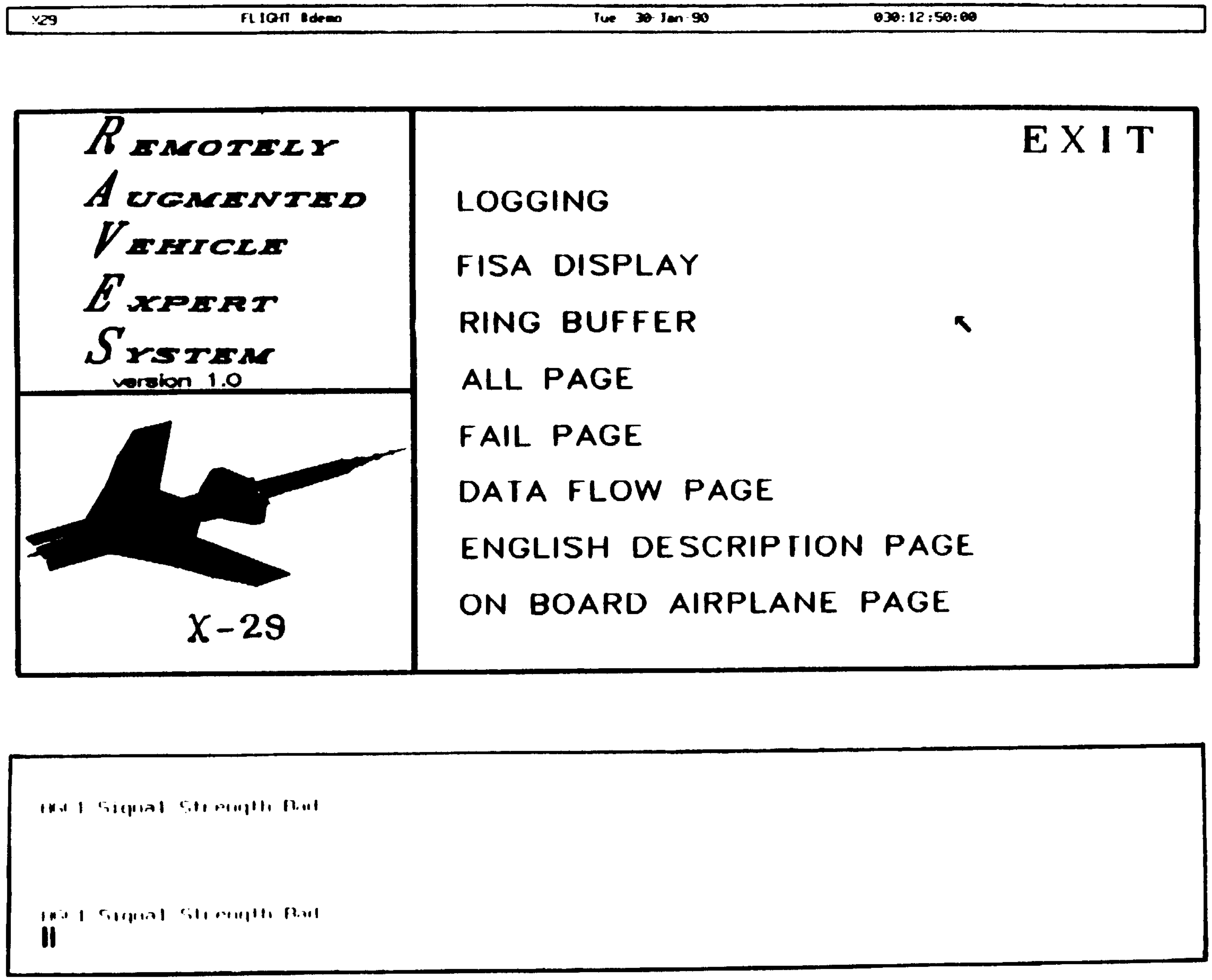


Figure 8-1. Top-level Display for RAVES (CSC, April, 1990)

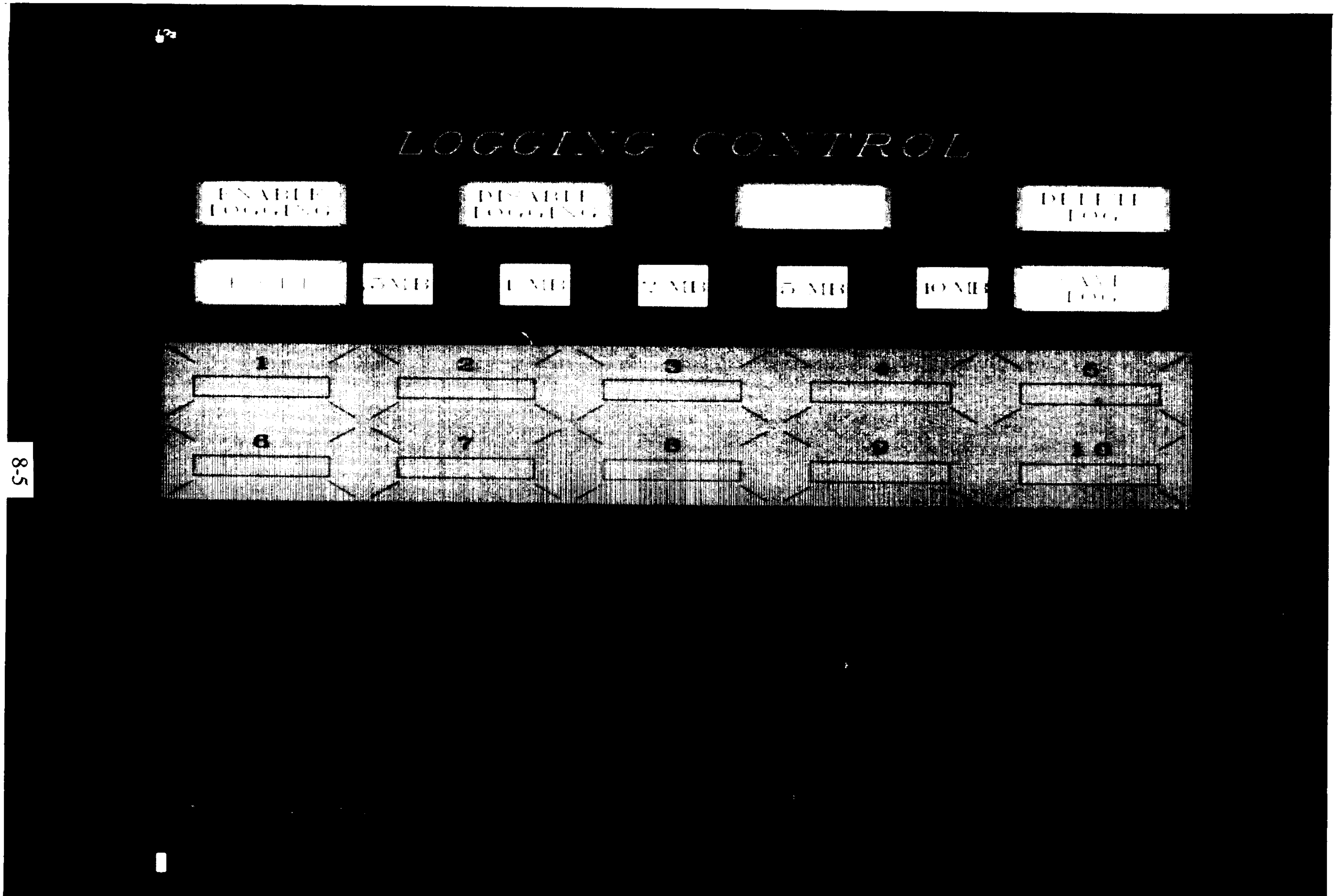


Figure 8-2. Logging Control Display from RAVES

ORIGINAL PAGE IS
OF POOR QUALITY

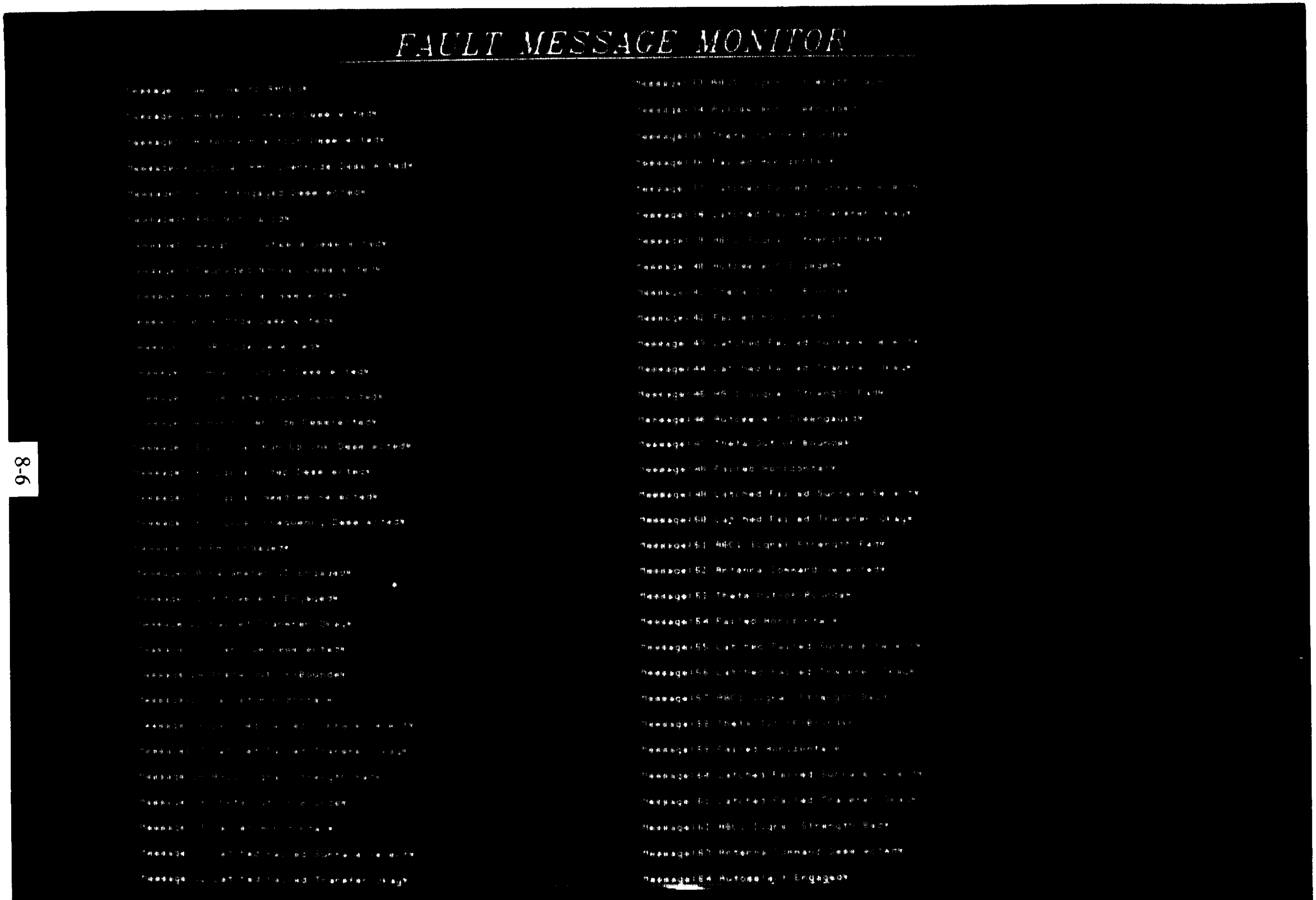


Figure 8-3. Fault Message Monitor Display from RAVES

4) All Page

This window displays the current status of a set of pre-defined parameters. Color coding is used to indicate the status of the parameter:

- Black: not failed, not selected, or nominal

  The meaning of black depends on the type of parameter. For discretes, black is off. For parameters that can fail, black is nominal or not failed.
- Green: good

  Green is used for parameters where it is important to determine that they are verified as "good", not just to determine that they are "not bad". Green indicates that the parameter has been checked for the proper value.
- Red: bad
- Yellow: failing
- Blue: selected (i.e., on, used for discretes that don't fail)

  Actual, dynamic data values are provided as supplementary information for some parameters. See figure 8-4 for an example screen.

5) Fail Page

This window shows the current status of a set of pre-defined critical failure parameters. The status is indicated using the following color coding:

- Black: non-failing
- Green: good
- Red: bad
- Yellow: failing

See figure 8-5 for an example of the Fail Page window.

6) Data Flow Page

This window provides a high level block diagram of the status of data flow in the RAV system. System failures are color-coded on the graphic. Examples of system failures visible from this graphic are equipment failure, software problems, and abnormal signal strength. Color coding used for this window is:

- Green: good
- Red: bad
- Yellow: failing
- Grey: assumed nominal; no actual data to support assumption

Selection of the aircraft in the schematic of this window causes display of the Aircraft Fault Detection window. See figure 8-6 for an example screen.

7) English Description Page

This window displays the same parameters as the All Page window, but uses an English descriptor instead of the abbreviated names used in the All Page. Color coding and window layout are the same as the All Page window. See figure 8-7 for an example screen.

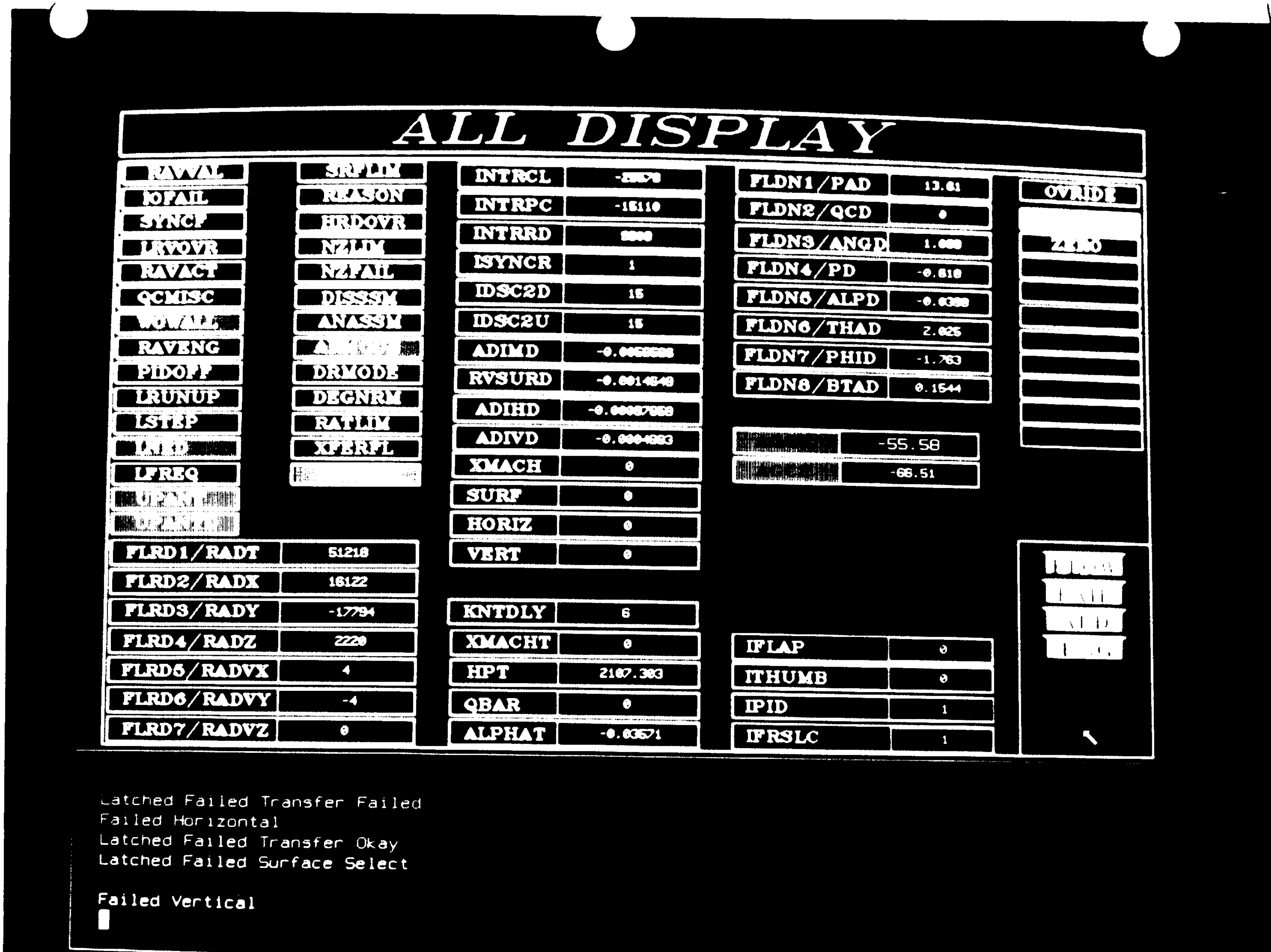


Figure 8-4. All Page Display from RAVES

ORIGINAL PAGE IS
OF POOR QUALITY

C-2

ORIGINAL PAGE
COLOR PHOTOGRAPH

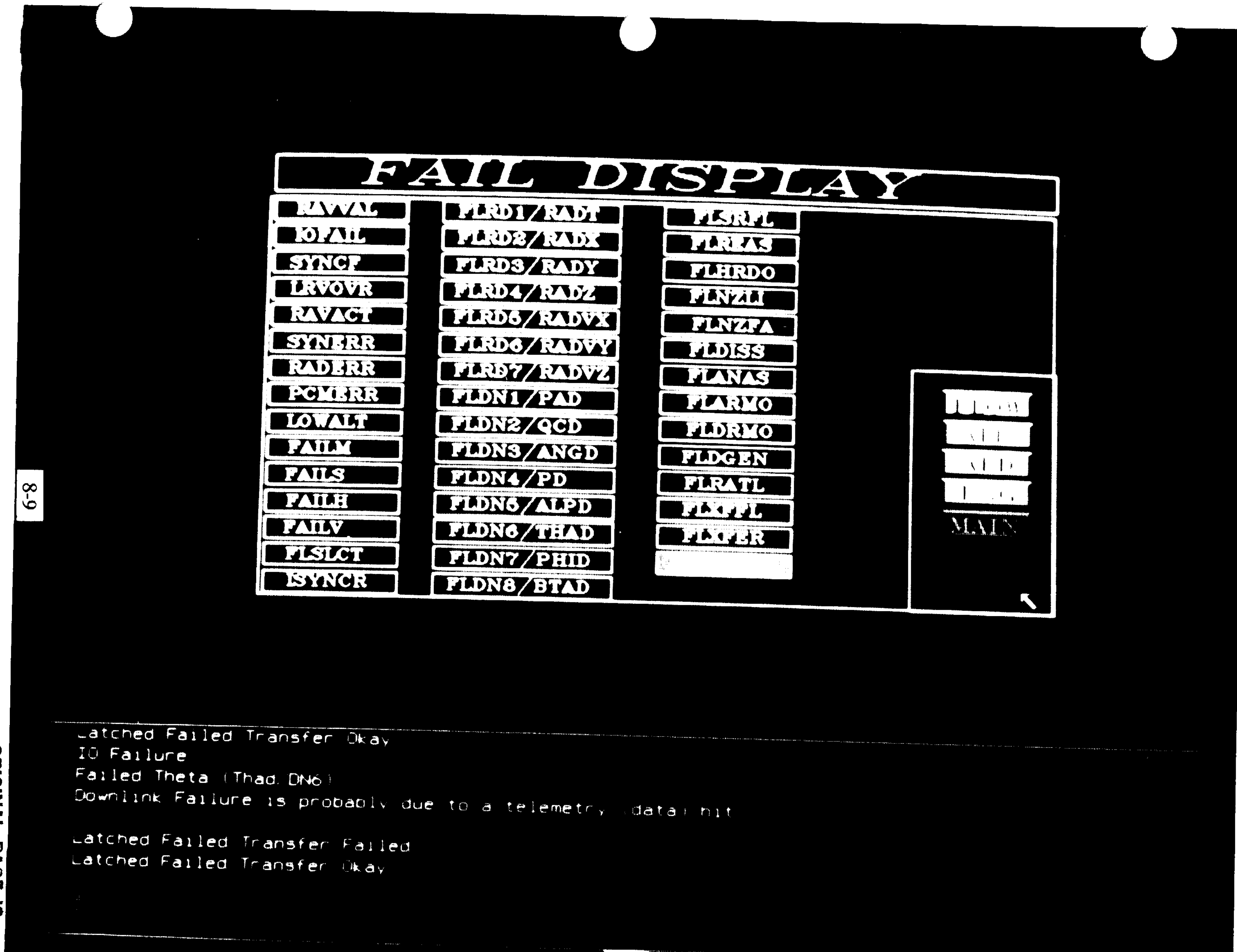


Figure 8-5. Fail Page Display from RAVES

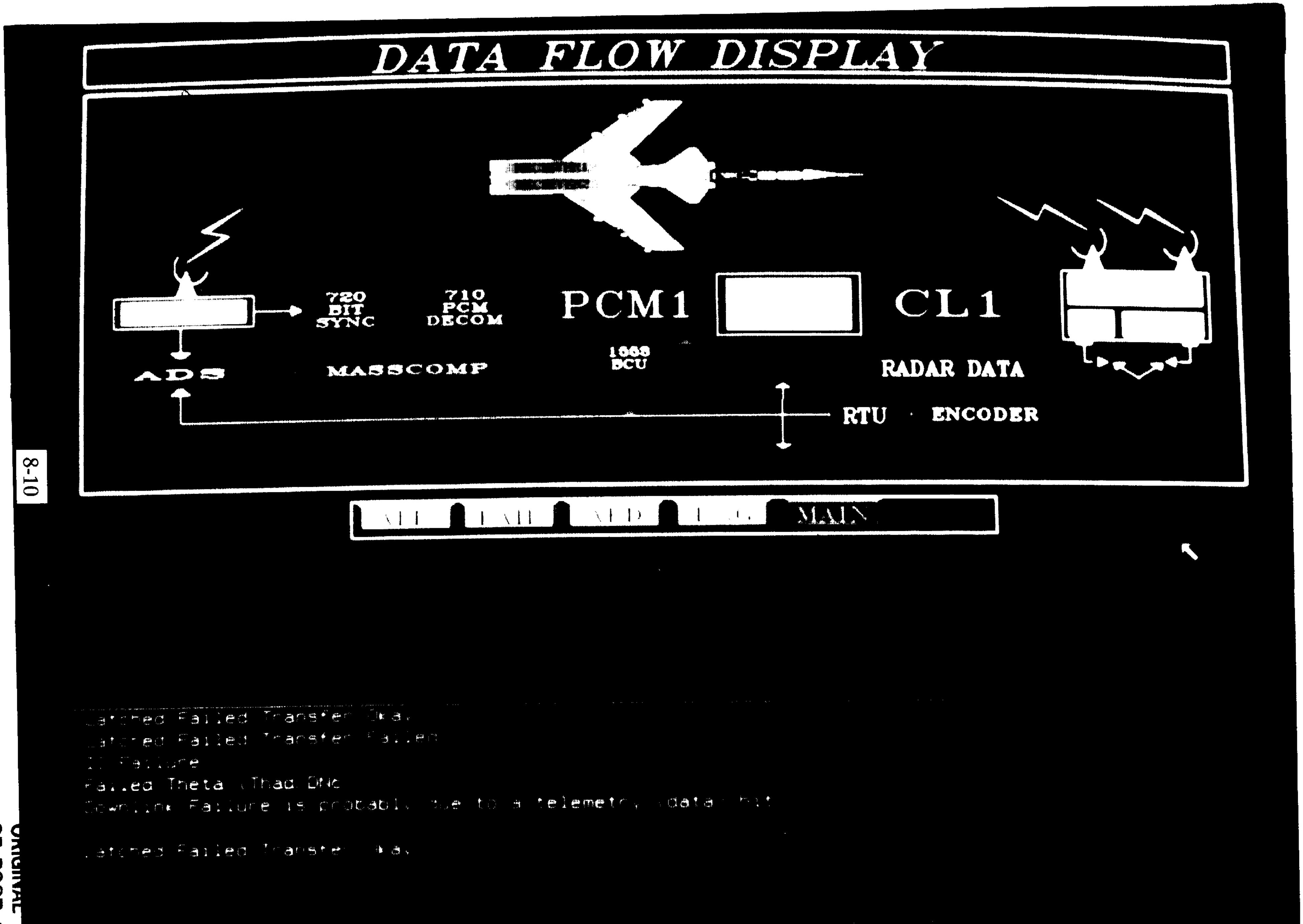


Figure 8-6. Data Flow Page Display from RAVES

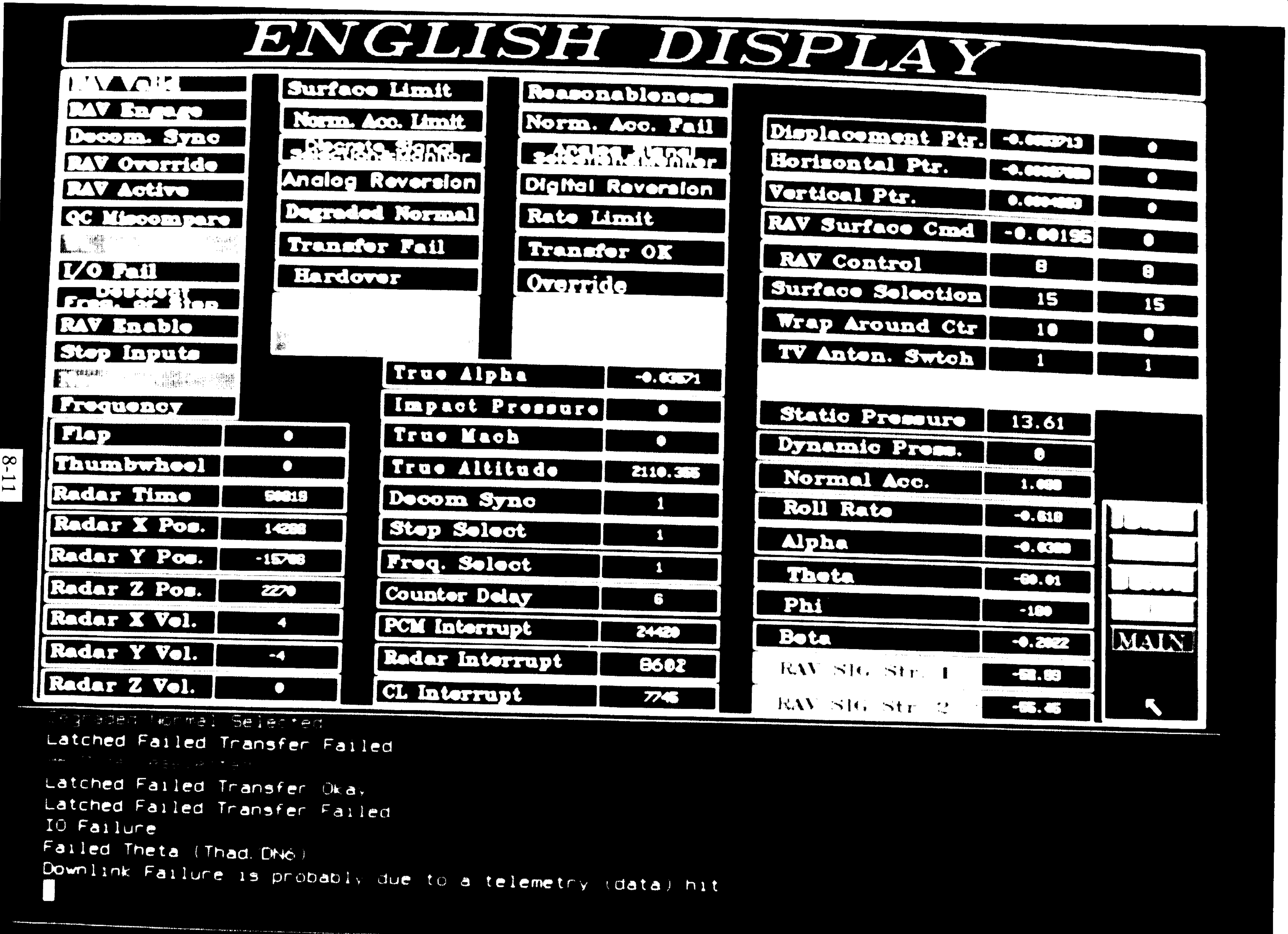


Figure 8-7. English Description Page Display from RAVES

8) On Board Airplane Page (Aircraft Fault Detection)

This window contains a schematic that illustrates a portion of the onboard electronic logic circuitry that received commands from the RAV. Downlisted telemetry drives the display. Colors used to indicate status are:

- White or green: good
- Red: bad
- Yellow: failing
- Blue: current surface

Information available from this display includes:

- Aircraft surface being pulsed
  Which effector is being used to implement control commands
- List of RAV commands with associated numbers
- RAV condition
  RAV condition must be engaged for control uplinks from the RAV to be used onboard. RAV condition can be manually set or automatically disengaged from onboard if the uplink has a potentially harmful effect ("fall out of RAV").

See figure 8-8 for an example screen.

Direct access of alternate views is available from the following windows:

- Data Flow Page
- Fail Page
- All Page
- English Description Page
- Aircraft Fault Detection

These windows also contain buttons for returning to the Main Menu and aborting processing.

The Expert Window is a window providing fault information from the expert system that pops up in the Main Display area when a "serious or unusual" error occurs. This window includes an explanation of errors and specifies required operator action. If actions involve multiple steps, the sequence is listed in order. The Expert Window is only available when the expert system is enabled. Other RAVES displays are active when the expert system is disabled.

The Fault Message Window is always displayed. Messages are color-coded and timetagged within this window and provide information such as flight status, system failures, and telemetry failures. Very serious (i.e., possibility of flight abort) messages are indicated using an audible tone. In the event of a very serious fault, the expert system (if enabled) provides error information and possible solutions via the popup window. Colors used to encode fault messages are:

- Red
  Serious fault, possibly flight critical
- Yellow
  Informational, not a flight critical fault
- Blue
  Information about the selected function; not flight critical
- Green
  Serious faults that have been corrected

Application fault messages are associated with specific variables. Any display containing these variables will include color-coding that reflects the content of related fault messages.

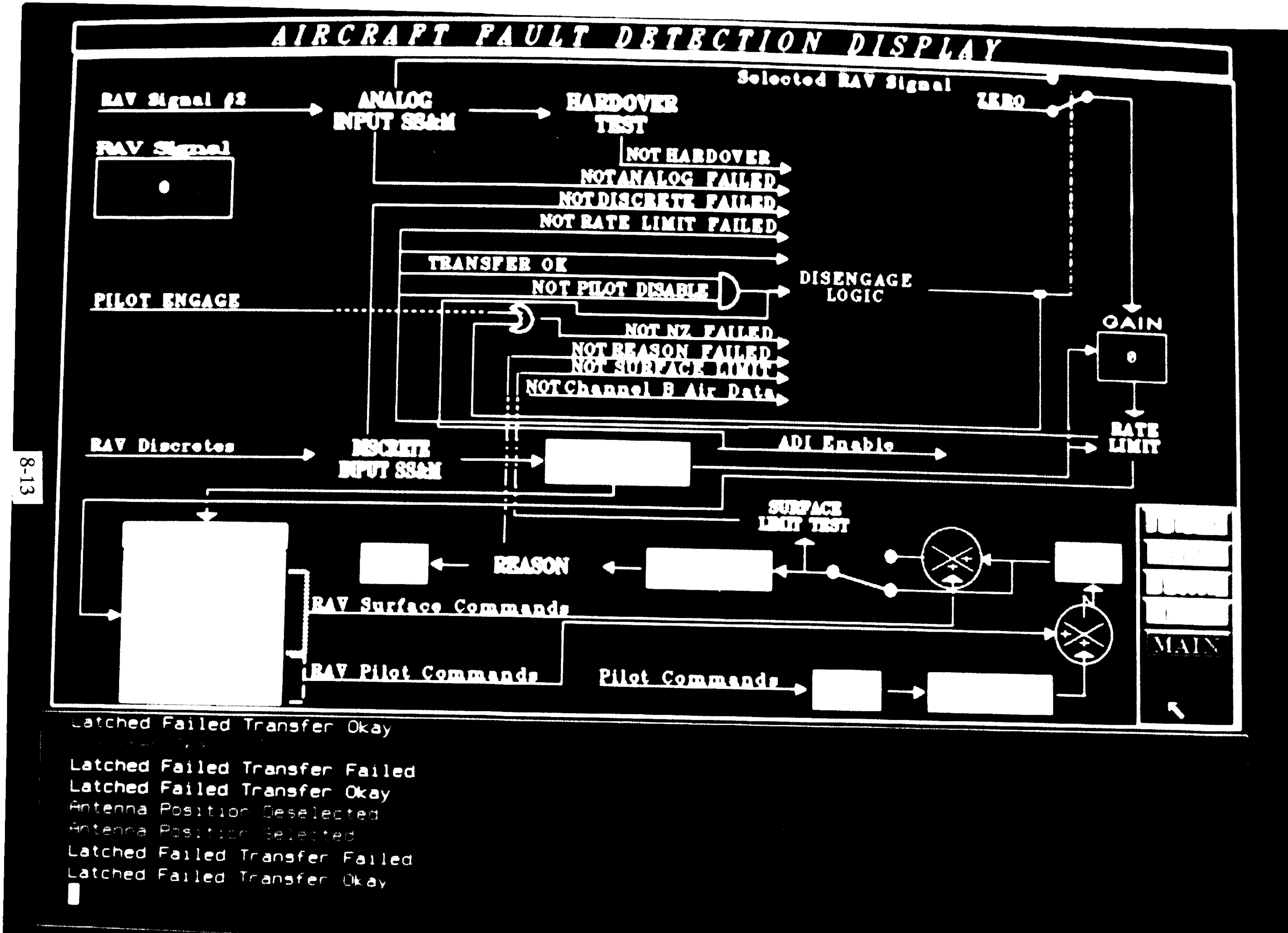


Figure 8-8 On Board Airplane Page Display from RAVES

8.5 Design Process

The RAVES system development philosophy includes user-designed displays. The selection of DataViews as the baseline interface tool was influenced by the need for a tool that was easy for users to learn. The procedure for generating a new display page is to develop the desired design using DataViews off-line, generate a list of the parameters to be displayed (i.e., a data source list), and provide both of these descriptors to the RAVES System Analyst. Software implementers then modify the operational system to include the new display.

8.6 Study Method

Information about RAVES was obtained by interview of the project representatives on April 30, 1990, and by review of the case data sources cited below.

Study Team

- Debra Schreckenghost (The MITRE Corporation)

Project Representatives

- Dorothea Cohen (NASA Ames Research Center)
- Dale Mackall (NASA Ames Research Center)

8.7 Case Data Sources

CSC (1990), *RAVES Programmer's Document*, Version 1.0, prepared for NASA by Computer Sciences Corporation.

CSC (January,1990), *RAVES Design Documents*, Version 1.0, prepared for NASA by Computer Sciences Corporation.

CSC (April, 1990), *Remotely Augmented Vehicle Expert System User's Guide*, TM-4000-04-01, Revision 1, prepared for NASA by Computer Sciences Corporation.

Mackall, Dale, David McBride, and Dorothea Cohen (May, 1990), *Overview of the NASA Ames-Dryden Integrated Test Facility*, NASA Technical Memorandum 101720, Edwards, CA: NASA Ames Research Center's Dryden Flight Research Facility.

Section 9
Real-Time Interactive Monitoring Systems (RTIMES)

9.1 System Description

The Real-Time Interactive Monitoring Systems (RTIMES) is a project to develop intelligent applications for real-time aircraft monitoring. The Short Take Off and Landing (STOL) Maneuver Technology Demonstrator was selected for initial prototyping. RTIMES includes three applications:

- Flight Path Control Set (FPCS) with the Discrete Monitor display
- Nozzle Controllers with the Nozzle Schematic display
- PROpulsion monitorinG Real-time Expert SyStem (PROGRESS)

Each application represents a different aspect of the same task. These applications have not been integrated. This was a test bed exercise only and there are no plans to use them operationally, therefore they never reached the integration stage. An initial testbed prototype was completed for the STOL effort. There are, however, plans to build an operational capability for the F-16 program that will be a continuation of this project. This effort has initiated and hardware has been purchased.

The RTIMES architecture, based on the RTDS software architecture, has three layers of information:

- Data acquisition
- Information gathering
- Knowledge application

Display of information occurs at the information gathering level. The user interfaces developed for these applications incorporate the following display concepts to achieve intelligent monitoring capability:

- Graphical displays
- Conceptual organization of displayed data
- Verbal descriptions of displayed data

The Real Time Data System (RTDS) hardware configuration, data acquisition software, and system design developed at the Johnson Space Center (JSC) were used with minor alteration (e.g., data rates of 100-1000 samples per second and screen update rates of 5-20 updates per second). The FPCS application is coded in C and does not use a rule-based knowledge representation. Both the Nozzle Controller and PROGRESS were developed using CLIPS. Most of the rules for the Nozzle Controller, however, are for the graphical display of information.

9.2 Intelligent System and Functions

Flight Path Control Set (FPCS)

The Flight Path Control Set application assists test engineers in monitoring the FPCS computer by interpreting discretes to determine the state and status of its components. Parameters with discrete values (i.e., 0 or 1) are downlisted as single bits in a 16 bit parent word. Each word represents a component of the FPCS computer. Typically, these discretes are manually interpreted by flight test engineers into state and status information by visually comparing the bit patterns formed by these discretes to documented templates (usually in a paper format).

The FPCS is the only application that does any message logging. It initiates limited logging of child messages when a bit is set (i.e., bit-triggered logging).

Nozzle Controller

The Nozzle Controller is a rule-based system that monitors and assesses state, status, and configuration information about the nozzle computer controllers and hydraulic vectoring actuators of the aircraft engine nozzle system. There are two nozzle controllers, one for each engine of the aircraft. Each nozzle controller has two redundant channels for actuator control. The intelligent system detects actuator failures and loss of a controller channel. It provides messages that describe faults indicated in the data discretes and alerts the test engineer to power down an engine when both channels of the engine are lost.

PROpulsion monitorinG Real-time Expert SyStem (PROGRESS)

PROGRESS is a rule-based system for monitoring the F-15 propulsion system during STOL. It monitors parameters indicating flight conditions, assesses the status of data acquisition, and detects engine anomalies by comparing temperature and pressure to the nominal operating limits. Text messages from the intelligent system provide two types of information: status of the engine and procedural recommendations based on that status. Safety critical recommendations (i.e, fail safe) are distinguished from other recommendations.

Noisy data was a problem for PROGRESS. Noisy data can misdirect the intelligent system and result in false alarms that are distracting to the flight test engineer. Capabilities provided to compensate for noisy data include:

- Allowing the operator to reset the expert system
- Implementing a primitive data filter to remove the effects of noise

The issue of noisy data is one that was encountered in other RTDS applications (e.g., Space Shuttle Payload Deployment and Retrieval System (PDRS) Operations Monitor). The ability to adapt to or compensate for noisy data is required of any system that monitors actual sensed measurements.

9.3 Human-Intelligent System Interaction Functions

Flight Path Control Set (FPCS)

The FPCS application assesses the status of the components of the FPCS computer by interpreting the discretes from the FPCS computer. The operator can select to display specific parent words with the status of each bit of that word. Anomalies are brought to the attention of the test engineer by highlighting the text description of the anomaly.

This application performs monitoring functions only. There is no ability to control the FPCS computer from this application.

Nozzle Controller

The Nozzle Controller monitors and assesses state, status, and configuration information about the nozzle computer controllers and hydraulic vectoring actuators of the aircraft engine nozzle system. A schematic of the controllers is used to present this information to the test engineer. Text messages describing faults indicated in discretes are also provided.

This application performs monitoring functions only. The test engineer cannot control the nozzle controllers from this application.

PROpulsion monitorinG Real-time Expert SyStem (PROGRESS)

PROGRESS monitors and assesses the engine status, flight conditions, and the status of data acquisition during STOL. It makes procedural recommendations based on engine status. It also detects the potential for an anomalous situation to occur as well as detecting the actual occurrence of the anomaly. Engine status and recommendations are provided as text messages, with color coding to differentiate between nominal and off-nominal status. An engine schematic is used to illustrate engine parameters that are operating out of limits. Strip chart emulations provide plots of important engine parameters.

PROGRESS is a monitoring application only. There is no ability to control the propulsion system from this application.

9.4 Supporting User Interface Capabilities

Flight Path Control Set (FPCS)

The Flight Path Control Set uses color coding to indicate status of components of the FPCS computer and provides on-screen descriptors of the discretes. Discretes are grouped into 16 bit integer words for downlisting. The complete 16-bit word (i.e., *parent* discrete) represents some component of the FPCS computer. Each bit or some combination of bits (i.e., *child* discrete) from this parent discrete represents conditions of the parent. The important information to display about a parent discrete is the setting of any constituent child discrete within the parent. Color-coded buttons labeled with the names of the parent discretes and organized into a panel are provided as a means of accessing information about the child discretes. The color conventions used for parent words are:

- Red
  The value of at least one of the children has been set to 1
- Blue
  No children have values of 1

Figure 9-1 provides an example of the Discrete Monitor display that is the interface to the FPCS computer. As seen in this figure, approximately 100 buttons are available for parent discretes. The unlabeled buttons were provided to allow the addition of new parent discretes as needed.

Tabular information about the particular children that have been set can be accessed by selecting the parent button. This information includes the name of the parent, the bit pattern of the parent (i.e., blank = not-set, 1 = set), and an interpretation of the meaning of the value of each child discrete (i.e., the child message). The child messages are coded as follows:

- Green
  Child discretes that are not set
- Yellow on Red Background
  Child discretes that are set, indicated with a 1 in the bit pattern at the top of the message panel

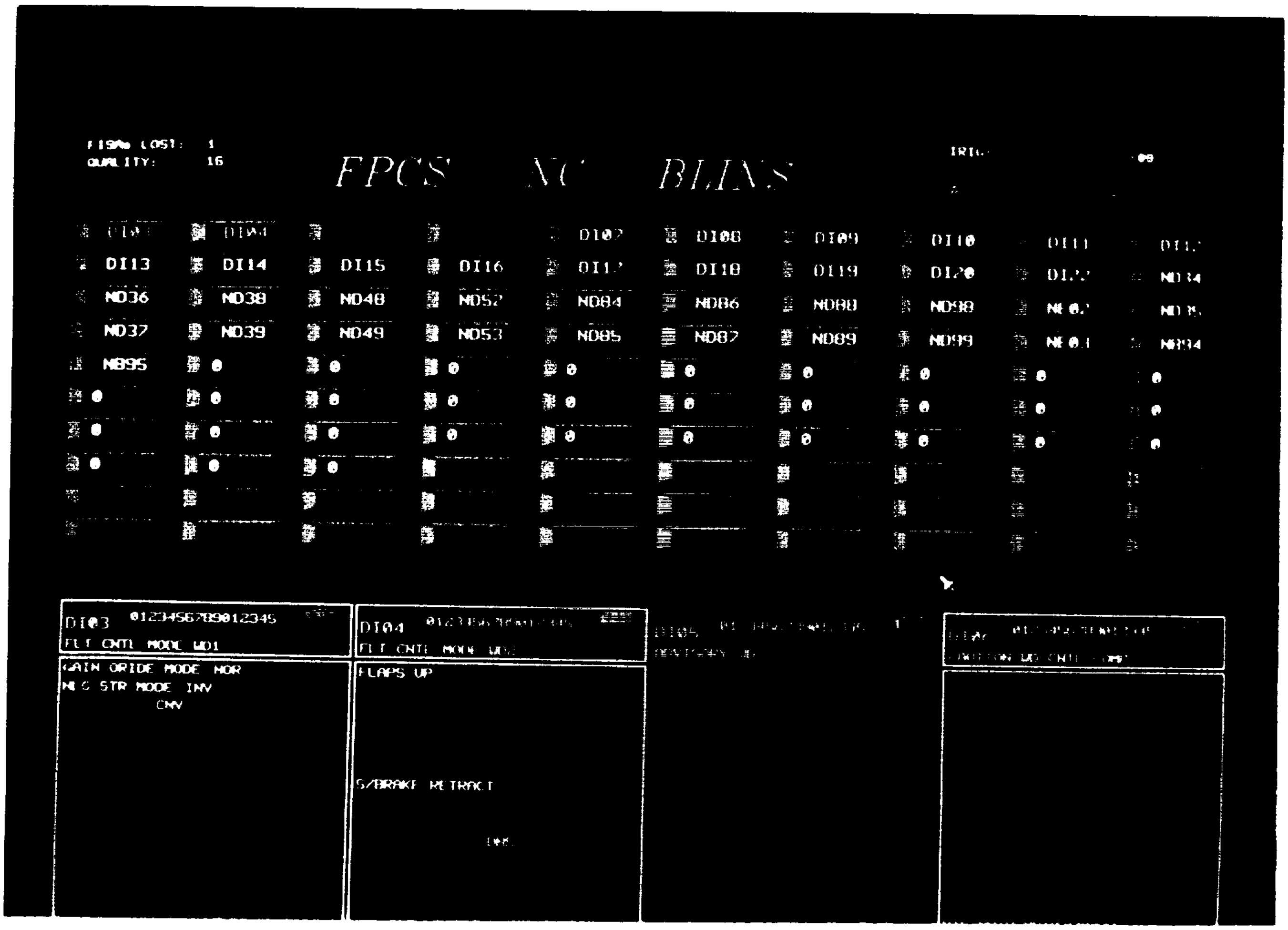


Figure 9-1. Discrete Monitor Display from RTIMES

ORIGINAL PAGE
COLOR PHOTOGRAPH

For example, in figure 9-1, the parent discrete DI04 contains two yellow status descriptors in the child messages (1) Weight-On-Wheels (W-O-W) Sel 0, (2) D-A-G Enabled OFF, and one red status descriptor in the child message D-A-G ENGAGED.

Nozzle Controller

The user interface developed for this application is the Nozzle Schematic (see figure 9-2), a wiring schematic that illustrates the current state and configuration of the nozzle computer controllers and hydraulic vectoring actuators (5 actuators per nozzle for both the left and right engines). Both nozzle controllers, with their redundant channels (A and B), are represented on the schematic. Connectivity and redundancy is illustrated as well as current status of the components of the engine nozzle system. Both engines are shown on this display, with the right engine above the left engine. Nothing is sensitive to user input on this screen.

Discrete information is provided within the blocks representing each component. These discretes indicate actuator failures and controller channel loss. For the controllers, the bit patterns are illustrated similar to the method used for the Discrete Monitor (i.e., blank = not-set, 1 = set, with "1" positioned above the bit number to illustrate set bits). There is a hierarchy of controller logic, where Nozzle Controller (NC) discretes are affected by both channels (i.e., A and B) to provide redundant monitoring. The actuator discretes within the five boxes can also be redundantly controlled by both channels. The windows to the right of the controller block show parent discretes within a specific channel (ND__). The larger window to the left that spans both channel blocks of the nozzle controller shows parent discretes affecting both channels (NC__). Within the actuator blocks, related discretes are also shown.

The wiring connections between the engines and actuators in the schematic are color-coded. Color indicates the state of the nozzle controllers:

- Light blue
  Nominal; indicates there are no failures in the components and redundant channels are each providing half of the current for actuators
- Green
  Loss of redundancy; indicates the remaining active channel after a channel has gone down and which actuator discretes are affected
- Red
  Fault; indicates loss of a channel and which actuator discretes are affected

A window containing messages describing each child discrete that has been set and each parent discrete that contains a set child is provided at the bottom of the screen. These messages explain all the fields highlighted in red in the schematic. The message format is:
parent bit# onboard_software fault_description

Loss of both channels results in a recommendation to power down the engine and a message "fail safe" is displayed on a red background above the blocks where the failures occurred. In figure 9-2, the right engine has lost both channels, resulting in a fail safe condition.

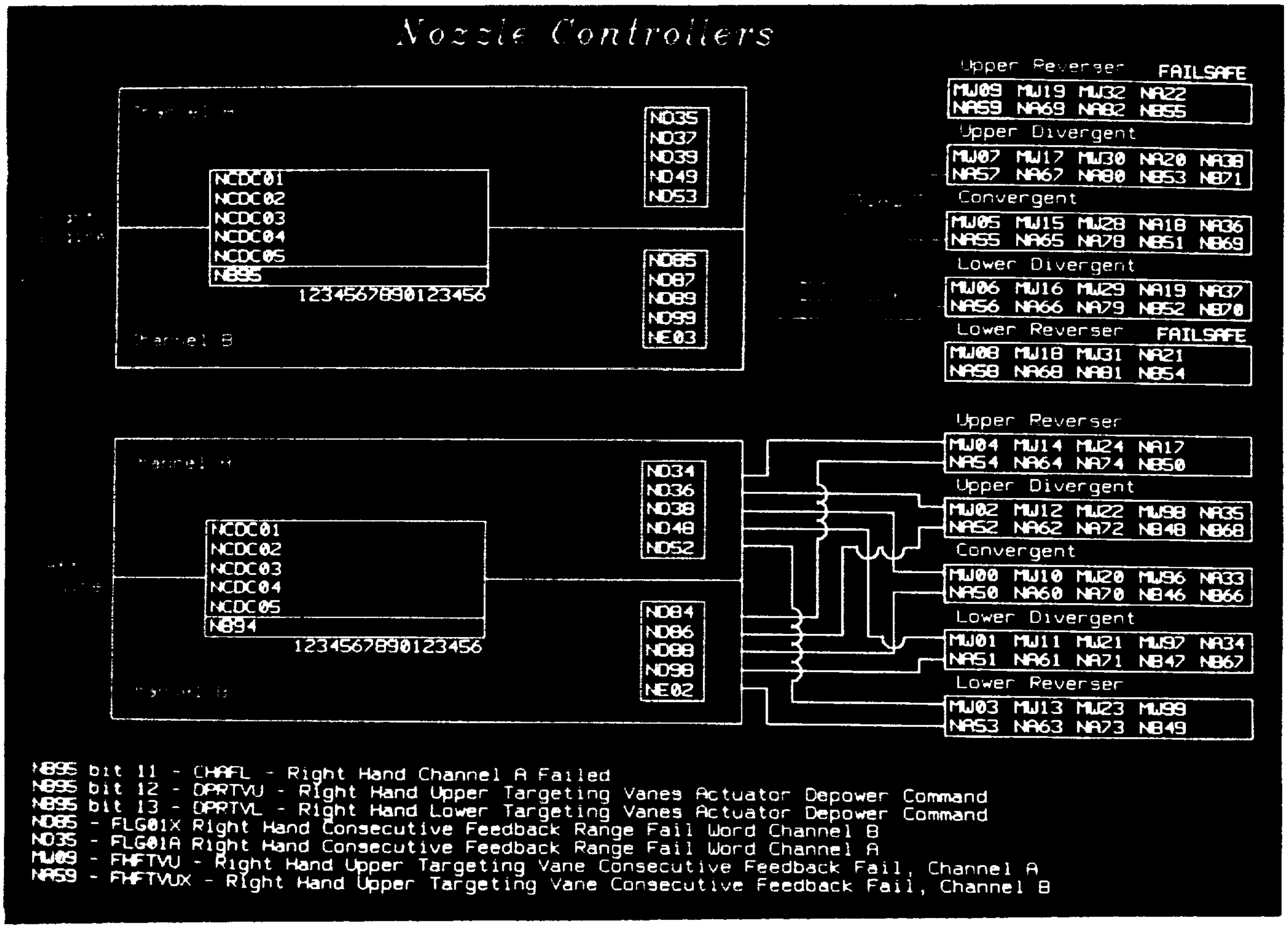


Figure 9-2. Nozzle Schematic from RTIMES

PROpulsion monitorinG Real-time Expert SyStem (PROGRESS)

The display workspace consists of four major regions, from top to bottom:

- Flight conditions and status of data acquisition
- Strip chart
- Expert system messages
- Engine schematics

The left engine is illustrated on the left of the screen and the right engine is illustrated on the right. See figure 9-3 for an example display.

The flight conditions are determined by monitoring a few key parameters:

- Altitude
- Mach number
- Alpha (i.e., flight path angle)
- Airspeed
- IFPC mode (i.e., vectoring mode of the turbine; possible values are conventional, reversal, afterburn)

Quality of the data acquisition process is indicated by the number of good frames per data cycle (1 cycle per second). The maximum number of frames per cycle is 16, so 16 is the best quality possible (i.e., no data lost).

The strip chart emulates an existing paper display capability for user-selected numeric parameters. Typical parameters plotted on the strip charts include turbine temperature (FTIT), power lever angle (PLA) to indicate engine power up, and core speed (NZ). The button labeled "strip charts" in the upper right hand corner is selected to define which parameters to plot. Selection of this button pops up a menu of parameters available for selection. Different parameters can be displayed for each engine. There is no color-coding used on the strip chart display. There has been some user dissatisfaction with this emulation due to the lack of a hardcopy capability. Apparently, a current technique is to review these hardcopies during postflight analysis. Although data are still accessible through data playback, users find this approach time costly.

The engine schematic shows a cross-sectional view of the engine. Key parameters to monitor during flight are temperature and pressure. Gauges showing these measurements are positioned on the schematic near the location where they were measured, with temperature above the engine and pressure below the engine. A thermometer-like display technique is used, where a bar marks the value on a vertical scale. Both predicted (left) and actual (right) measurements are shown on the gauge. The exact value of the actual measurement is shown in a color-coded panel near each gauge as well. A value within the nominal operating limits is displayed in white text on a black background. When a value is outside of the normal operating range, it is displayed as white text on a red background. Since the vectoring nozzles of the engine are closely monitored during STOL, the schematic changes physical appearance to illustrate the appropriate configuration for the end of the engine. The vectoring angle of the nozzles and reverser vanes are indicated using red bars.

The expert system provides two types of information: status of the engine and procedural recommendations related to that status. These assessments are displayed in the form of text

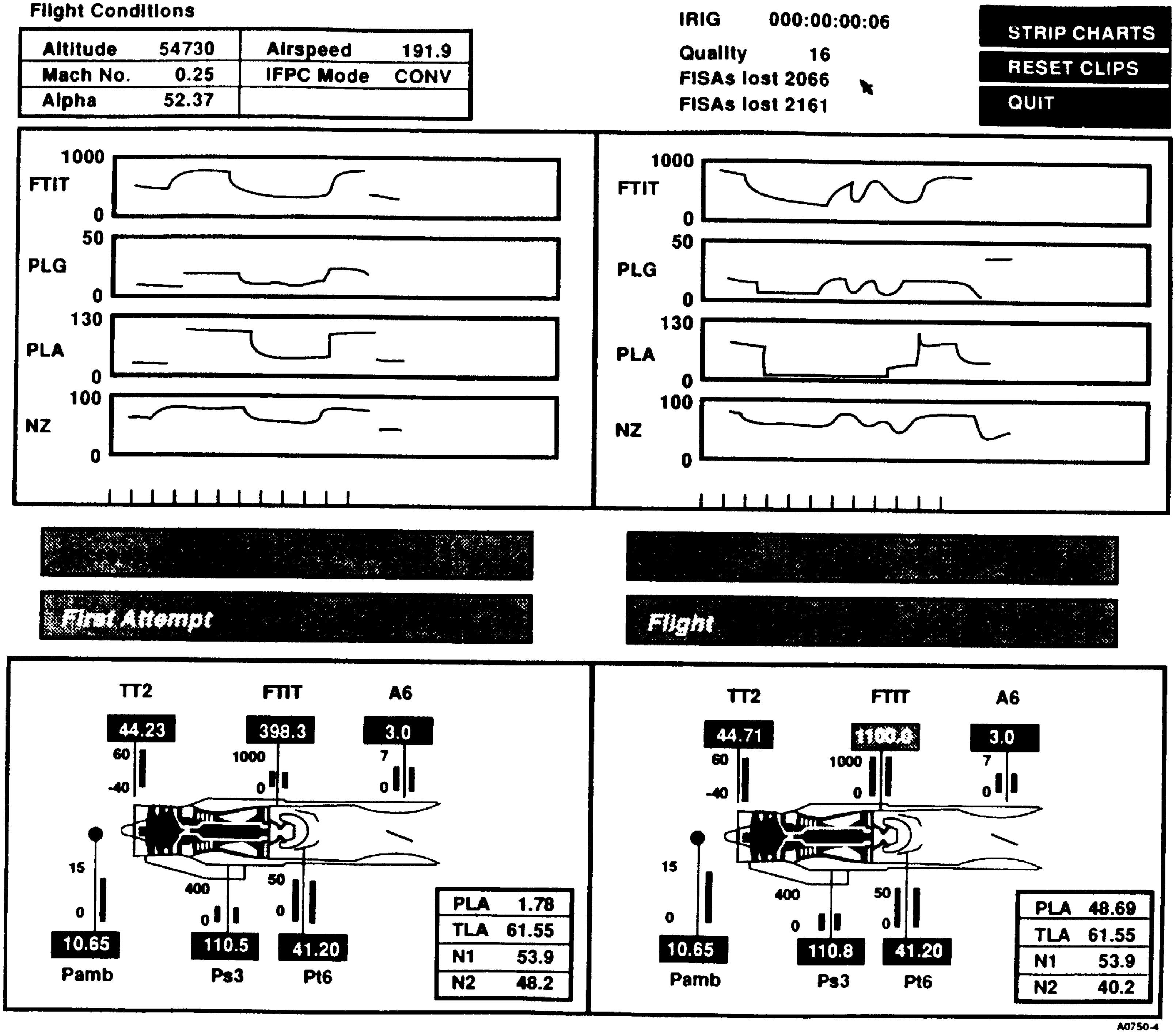


Figure 9-3. PROGRESS Display from RTIMES

messages, with the lower message for status and the upper message for procedures. Text messages are also color-coded for quick scanning:

- Green
  Normal engine status (e.g., good start)
- Yellow
  Potential anomalous condition (e.g., potential hot start)
- Red
  Anomalous condition has occurred (e.g., actual hot start detected)

Some states (e.g., "stall") are described as being easy to detect once they occur (i.e., red status) but difficult to predict the potential to occur (i.e., yellow status). The current system would only have a red status for these states. Operator control of the expert system consists of resetting CLIPS and quitting the application.

9.5 Design Process

The development process is described as "the continuous iteration process of test, evaluate, and modify" (Flanders et al, 1990). The test portion of this process proved critical. During testing, they discovered "hidden assumptions" that invalidated their rule base. PROGRESS was developed by a knowledge engineer and reviewed by a domain expert. Techniques used to elicit domain knowledge include a knowledge matrix generated by the knowledge engineer and a flow chart of monitoring activities generated by the domain expert. The initial rule set was based on these two forms of knowledge.

9.6 Study Method

Information about RTIMES was obtained by interview of the project representative Robin Madison on April 24 and August 6, 1990, and by review of the case data sources cited below.

Study Team

- Debra Schreckenghost (The MITRE Corporation)

Project Representative

- Robin Madison (Edwards Air Force Base)

9.7 Case Data Sources

Flanders, J.B., C.H.Jones, and R.M.Madison (May ,1990), "An Expert System for Real-Time Aircraft Monitoring", reprint from Proceedings of AIAA 5th Biannual Flight Test Conference.

Section 10
Onboard NAVigation (ONAV) Expert System

10.1 System Description

Two independent navigation state vectors are maintained for the Space Shuttle during all phases of flight, one determined by the onboard flight navigation software and one determined by ground-based navigation software. The Entry ONAV Expert System is an expert system built for support of the Onboard NAVigation (ONAV) ground flight controllers in monitoring the onboard navigation software and the sensors that provide the measurements used by that software during re-entry and landing of the Space Shuttle.

There are actually four prototypes planned for the ONAV flight controllers, corresponding to the phase of the mission that they will support (i.e., ascent, rendezvous, deorbit, and entry). Currently, the entry system is considered complete and is being tested in the Multi-Purpose Support Room (MPSR), although changes are expected in the system after being installed in the MPSR. The rendezvous system is in development and the ascent system is still considered a prototype. No work has been done on the deorbit system. This report describes the Entry ONAV Expert System.

One difference between all of the ONAV Expert Systems and the RTDS systems is the source of data. The ONAV Expert Systems rely on the Mission Control Center Upgrade (MCCU) Local Area Network (LAN) system for data while the RTDS systems have an independent telemetry distribution system. ONAV uses two types of data: trajectory parameters sensed or computed on the ground and downlisted telemetry parameters.

All ONAV Expert Systems were developed in CLIPS with supplemental C code to store and pre-process telemetry data. The delivery platform is a Masscomp workstation, but SUN® workstations have also been used for development. The entry expert system consists of approximately 300 rules. The user interface is written using Masscomp Graphics. The final user interface tool will be the X Window System™. The port to the X Window System has already been implemented. This version cannot be used in the MPSR, however, until the X Window System is available under the MCCU software configuration manager (i.e., the MCCU Workstation EXecutive (WEX)).

The user interface to the Entry ONAV Expert System has been ported multiple times. User interface ports include:

- CURSES to SunTools™
- SunTools to Masscomp Graphics
- Masscomp Graphics to the X Window System

In the course of porting the system, problems with the original CURSES interface were addressed, including the need to separate events from recommendations and quality from status. These had been combined originally due to limited screen space.

® SUN is a trademark of Sun Microsystems, Inc.
™ The X Window System is a trademark of MIT.
™ SunTools is a trademark of Sun Microsystems, Inc.

10.2 Intelligent System and Functions

The Entry ONAV Expert System is a rule-based system that assists flight controllers in assessing the quality of measurements from multiple navigation sensors, the compliance of the navigation state with flight rule limitations, and the quality of the navigation state vector computed onboard the vehicle with respect to the redundant ground-computed state vector. It provides information about the state of the power switches for sensors, which sensors have been selected when redundant capability is provided, configuration checks prior to sensor processing, and quality assessment of sensor data. The operator manually configures the expert system for mission support. This configuration information is typically only available on voice loop.

The intelligent system provides a variety of types of messages, including off-nominal events, detailed subsystem messages (e.g., messages affecting a specific type of sensor) and recommendations. Each message has a field where current vehicle altitude is displayed, since altitude indicates the regime of the entry phase, which defines what sensors are available, what activities are expected, etc.

The ONAV Plot System executes in parallel with the ONAV Expert System. This system plots parameters of interest with respect to altitude to detect significant disagreement that would require corrective action (e.g., uplink of ground state vector). Constraints from the flight rules are explicitly represented on these data plots (e.g., the conditions under which the redundant ground state is uplinked to correct the onboard navigation state vector). A planned enhancement of the ONAV Plot System is to indicate the data quality of a parameter on plots of that parameter by only plotting data with good data quality.

Telemetry data are pre-processed prior to use in the ONAV Expert System to convert to symbolic representation and to filter out noisy or erroneous information.

10.3 Human-Intelligent System Interaction Functions

The Entry ONAV Expert System assists the ONAV flight controllers in monitoring and assessing the state and status of the onboard navigation software and the sensor measurements used by that software. It presents information as message lists, tables, and panels of status words. The operator enters information using radio buttons and menus. The ONAV Plot System is also provided to assist controllers in comparing the onboard navigation state to the redundant ground navigation state. It presents information as data plots.

Support for collaboration includes two scrollable message lists. One list identifies off-nominal events. The contents of the other list are determined by the operator, who can select the type of messages to be reviewed depending upon the current conditions or phase of operation. For example, when baro data are being processed by the navigation software, messages about the baro system would be displayed. Data are logged for playback and review, but no playback capability is supported during real-time operations.

No capability to intervene with or control the monitored process (i.e., the generation of the onboard navigation state vector) is provided.

In addition to use during real-time operations support (both during missions and integrated training simulations), the Entry ONAV Expert System will be used for off-line training.

10.4 Supporting User Interface Capabilities

One workstation is available for displaying results from the Entry ONAV Expert System. The ONAV operator interface provides two full screen display formats, selectable by the operator. One display format shows status assessments, messages, and recommendations. The other display format shows plots of data values, in groups of 6 plots per screen. At the top of both screens is a configuration region showing Greenwich Mean Time (GMT), Mach number, and altitude. When running under WEX, a WEX menu region is displayed above the configuration region.

Expert System Display

The ONAV Status Screen is the display format that would normally be viewed during mission support. It provides a summary of current system state and status. Formats used to display this information include color-coded light panels with redundant text coding and text messages. Popup windows that overlay the configuration portion of screen are used for operator inputs. Schematics are not used, since they are not currently used in any form by the operators.

The operator must configure some system parameters manually. This is required for information not available via LAN (i.e., information only available on the voice loop). Operator inputs include:

- Landing runway specification [e.g, runway 17 at Edwards Air Force Base]
- Atmospheric conditions [i.e., nominal, hot, cold]
- Tactical Air Command and Navigation System (TACAN) [i.e., primary, secondary]
- Backup Flight System (BFS) [i.e., go, no go]
- Delta State Update: uplink of the redundant state vector computed on the ground
- High Speed Trajectory Data (HSTD) state vector quality: quality of the redundant state vector computed on the ground

Expert system control capabilities include:

- Return to CLIPS
- Reset CLIPS rule base
- Go (i.e., start the system)
- Exit

The screen is divided into regions of related status. These status regions are:

- System Power
  State of power switches for sensors
- Sensor Selection
  State of selection of redundant sensors
- Measurement/Processing
  Hardware Built-In Test Equipment (BITE) status, software enable flags, quality of redundant ground solution
- Quality Rating
  Expert system assessment of sensor quality

Additionally, two scrollable message windows are available, one for off-nominal events and the other for subsystem messages and recommendations. Operator inputs are provided via the System Configuration window and the Subsystem Options popup window. See figure 10-1 for the workspace and figure 10-2 for an example of the Entry ONAV Expert System Display.

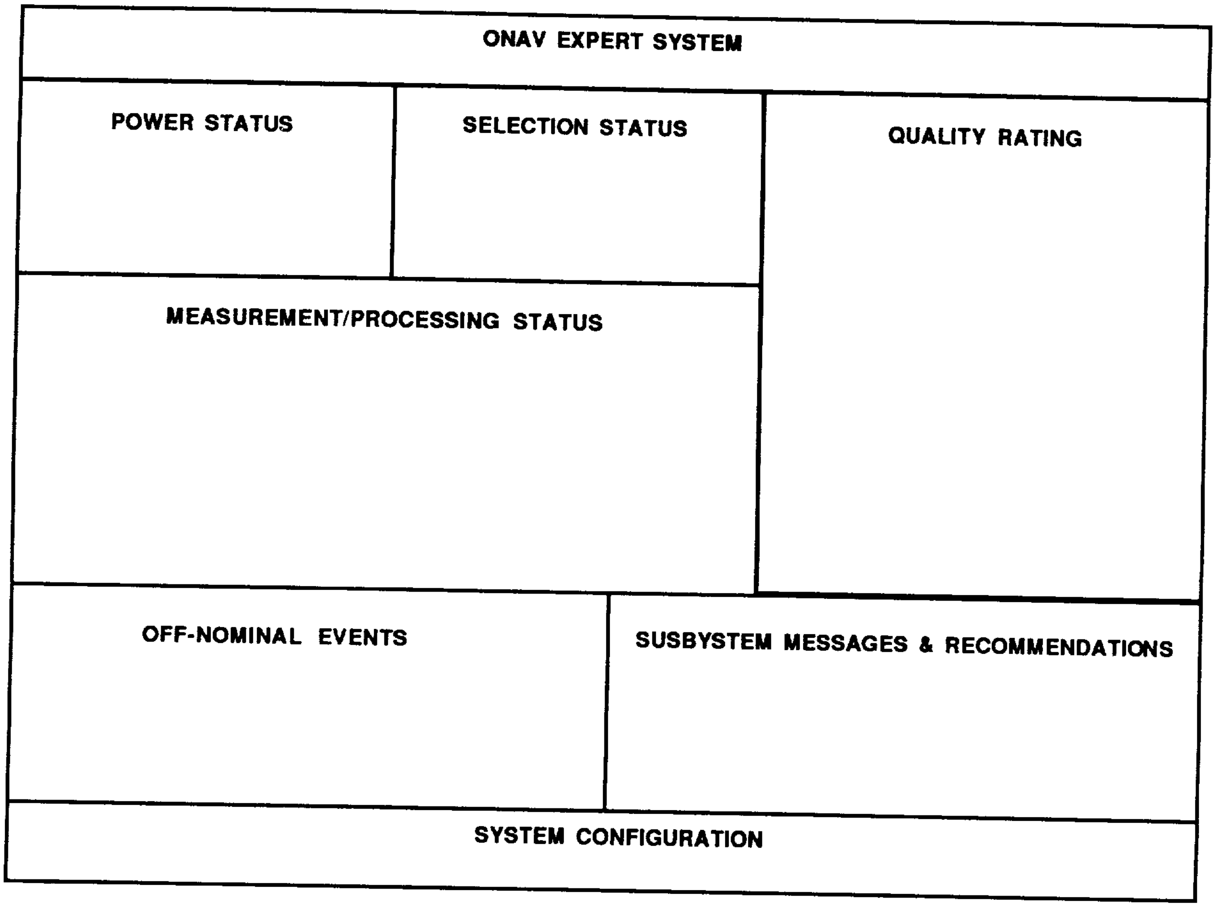


Figure 10-1. Workspace for the ONAV Expert System Display

Status regions are displayed on a blue background. Color used in the status regions are:

- Red - warning
- Green - good
- Yellow - caution
- Fuchsia - sensor data error (e.g., bias)
- Black with white text - sensor data invalid (e.g., bad air data due to roll reversal)
- White with no text - outside regime of mission where this status is relevant

The Quality Rating region displays quality assessments generated by the expert system. These assessments apply to the state of the Primary Avionics System Software (PASS), the BFS, the ground navigation system (GND), and available sensors (e.g., TACAN, baro, Inertial Measurement Units (IMU), Microwave Scanning Beam Landing System (MSBLS)). Three Line Replaceable Units (LRUs) are available for redundant sensor measurements and three entry navigation states are maintained simultaneously. Off-nominal status is also reiterated in messages (e.g., sensor measurement bias magnitude is provided in messages).

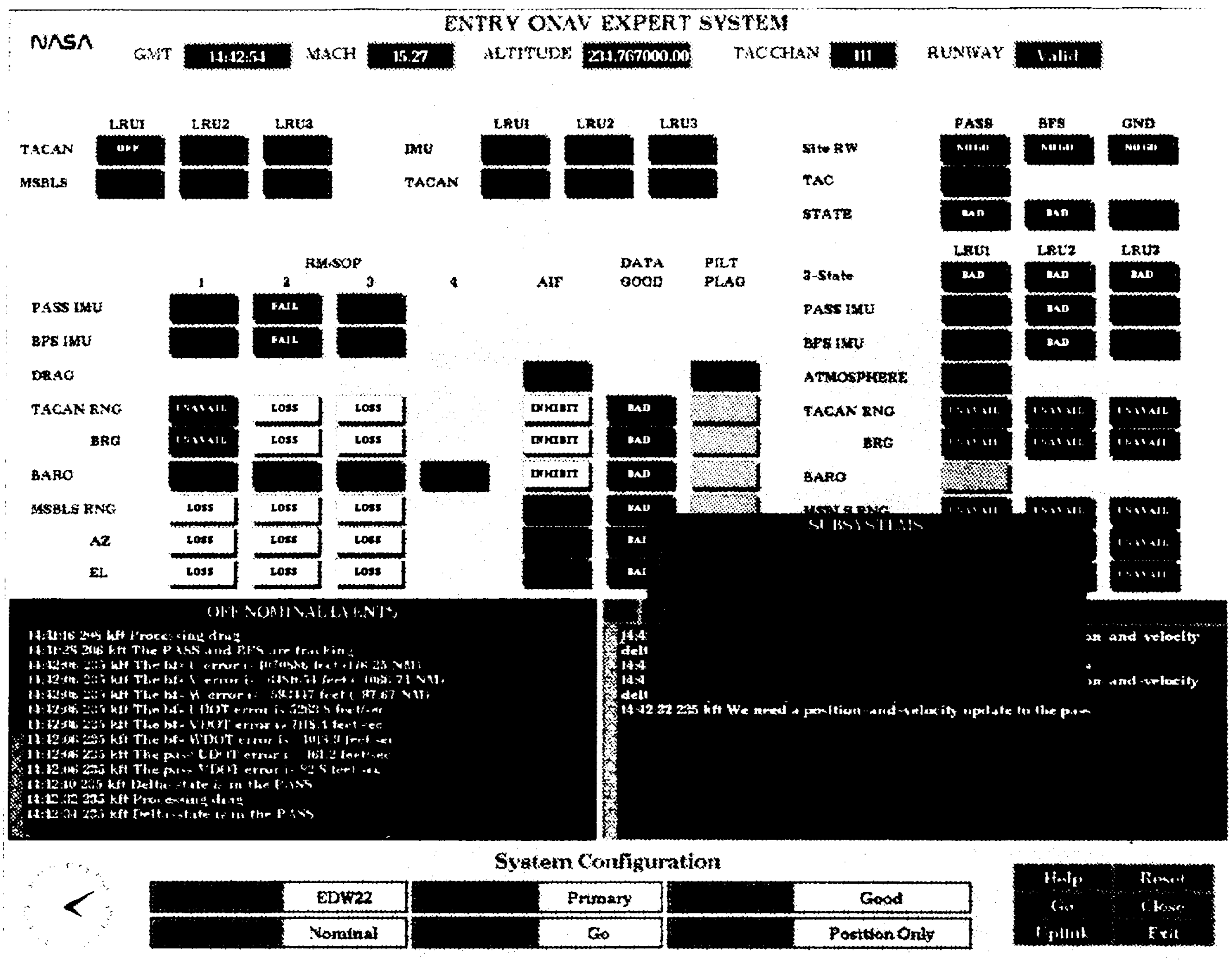


Figure 10-2. Entry ONAV Expert System Display

ORIGINAL PAGE IS
OF POOR QUALITY

Two scrollable message lists are displayed at the bottom of the screen:

- Off-Nominal Events
  Messages in this list notify the operator of all significant events, relative to entry navigation. This window is located in the lower left hand side of the screen.

- Subsystem Messages and Recommendations
  The types of messages displayed in this list are selected by the operator. The operator may choose to view more detailed subsystem messages (e.g., TACAN messages, state messages) or a summary of recommendations made by the expert system. Selection of the icon in title bar above the message panel pops up a menu for selecting the desired type of messages. A check mark to the left of the menu option indicates the selected subsystem. This window is located in the lower right hand side of the screen.

The message format used for both lists appends the time of the message and the altitude of the vehicle at that time to the text of message. Although the use of timetags with message lists was encountered in other applications, the use of altitude is unique to this application. It is provided since altitude indicates the regime of the entry phase, which defines what sensors are available, what activities are expected, etc..

Both events and recommendations are logged to file for off-line use. Recommendations are displayed only until they are satisfied. Orange text on black background is used for off-nominal events and green text on black background is used for subsystem messages and recommendations.

ONAV Plot Display

The differences between the onboard and ground state vectors (called a *state vector comparison*) with respect to altitude are monitored to detect significant disagreement that would require corrective action (e.g., uplink of ground state vector). The ONAV Plot System provides data plots of these state vector differences, comparisons of redundant sensor data, and the differences between measurements and state during Space Shuttle entry. This system was developed to execute in parallel with the ONAV Expert System.

The Data Plot display format consists of three full-screen pages with 6 plots per page. These 18 plots can be selected from 45 available parameters:

- State vector comparison for BFS
- Three LRU comparison for MSBLS
- Three LRU comparison for TACAN
- Three LRU comparison for IMU velocity (including activity threshold)
- Three LRU comparison for IMU attitude (including activity threshold)
- Three LRU comparison for IMU accelerometer components (i.e., x, y, z)
- Drag residual

All parameters are plotted with respect to altitude. Both ground-computed and onboard altitude are available for display. Using a popup menu, the operator can enlarge a plot to 2/3 screen size for detailed viewing (i.e., zoom) or change the page viewed. The page currently displayed is identified by a page number at the top of the display. Plots have fixed scales for all axes. Color-coding distinguishes the PASS state from the BFS state. See figure 10-3 for an illustration of the Data Plot display format.

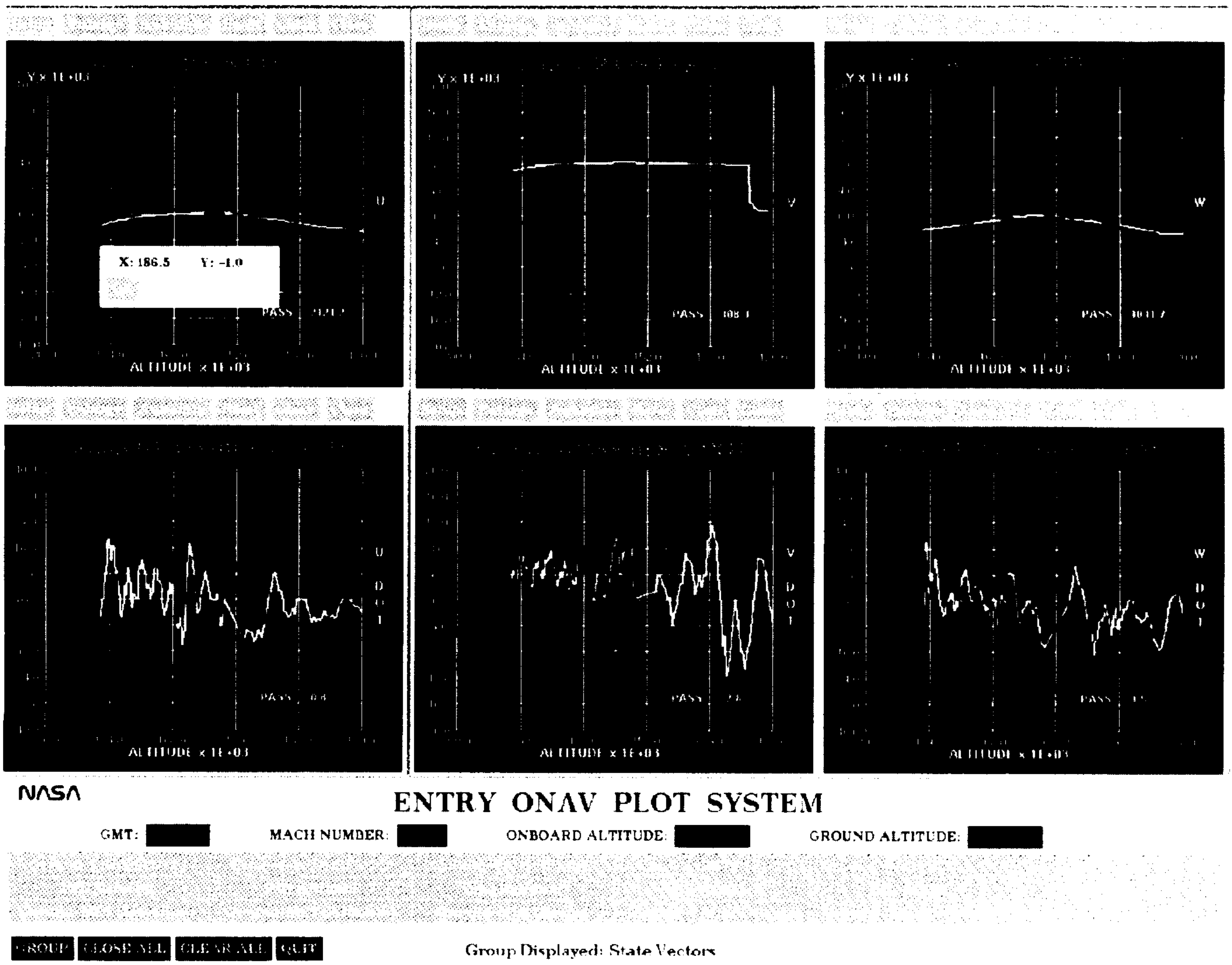


Figure 10-3. ONAV Plot Display

Information from the flight rules are included on these plots. Flight rules specify the conditions (i.e., altitude versus state vector error) under which a delta state update occurs. These flight rule limits are marked in red on the plots, so regions where state vector error exceeds these limits can be easily identified. Yellow means suspect limits.

Currently, data are plotted if is are available. A planned enhancement is to only generate plots when the sensor data used to determine the state vector are assessed as good. If the data are bad, nothing would be plotted. Another planned enhancement is to provide additional plots of other types of information.

10.5 Design Process

Unlike the RTDS applications, the all ONAV Expert Systems are being developed by an organization separate from the user community. Requirements were generated in working groups and iterative prototyping was used to refine the prototype into the completed system. No formal requirements were documented other than level A requirements generated after the prototyping effort (JSC, 1988). The expert system was developed first, then the user interface was developed.

A basic assumption in building the interface was that the old display formats (i.e., Manual Select Keyboard (MSK) and Digital Display Devices (DDD)) would be retained. The current MSKs and DDDs have been emulated in the X Window System for use in certification and testing of the expert system. Side-by-side comparison of the two displays will be performed during testing. Two physical terminals should be available for ONAV support in the final MCCU configuration. One would be used to display the expert system screens and one would be used to display the emulation of the current MSKs. They also plan to use the MSK during playback of data. Playback capability is currently in work.

The Entry ONAV Expert System was resident in the MPSR at the time of the interview. The development team feels that it is essential to have the system in the MPSR for operational testing to gain operator confidence in the system. This system is currently being tested by flight controllers during training simulations.

A certification process has been specified for all ONAV Expert Systems, which includes the following steps:

- Certifying the data acquisition software (i.e., software written in C to strip data from LAN, load it into shared memory, and pre-process it prior to expert system processing)

- Certifying the rule base using test cases that satisfy a pre-specified matrix of errors; the system must function correctly on at least two test cases for each specified error

- Certifying the user interface on the integrated system; this must wait until the X Window System is available for use in the MPSRs

- Final certification will require providing real-time support successfully during an entire simulation session; for entry, this is approximately 12 simulations.

The certification process is intended to involve all flight controllers in use of the expert system during training simulations. Operators log (i.e., write down) anomalies encountered during use and these anomalies are discussed later with system developers to clarify the problem. The Entry ONAV Expert System was expected to be certified during first quarter of 1991. The rendezvous portion of the expert system was expected to move into the MPSR in this same time frame. The ascent prototype is still in development.

10.6 Study Method

Information about the Entry ONAV Expert System was obtained by interview of the project representatives and demonstration of the prototype on May 16, 1990, and by review of the case data sources cited below. Two participants in the development process conducted the demonstration: Lui Wang from the development organization and Malisse Haynes from the user organization.

Study Team

- Debra Schreckenghost (The MITRE Corporation)

Project Representatives

- Lui Wang (NASA Johnson Space Center)
- Malise Haynes (NASA Johnson Space Center)

10.7 Case Data Sources

JSC (September, 1988), *Knowledge Requirements for the Onboard Navigation (ONAV) Console Expert/Trainer System*, JSC-22657, Version 1.1, Mission Planning and Analysis Division, Mission Support Directorate, Johnson Space Center, Houston, TX: NASA.

JSC (April, 1990), *Level-A Requirements for the Onboard Navigation (ONAV) Plot System*, Information Systems Directorate, Johnson Space Center, Houston, TX: NASA.

JSC (January, 1991), *The Real-Time X-Based Plot System Using a Generic Plot Widget*, Information Systems Directorate, Johnson Space Center, Houston, TX: NASA.

Section 11
Rendezvous Expert System II (REX II)

11.1 System Description

REX I was developed for support during the Space Shuttle Rendezvous phase and REX II for Space Shuttle Proximity Operations (Prox OPS) support. These phases correspond to a set of timeline activities with specific flight rules. Thus both are Space Shuttle applications, but include provision for docking with Space Station. REX I initiates when rendezvous navigation processing is enabled (60-80 nautical miles (NMI) from target) and provides continuous support through the final mid-course correction maneuver (MC4, ~3000 feet from target). REX II initiates at MC4 and provides continuous support through docking.

REX II represents an enhancement of REX I. This report documents REX II only. Hereafter, the acronym REX will be used to refer to the latest version of this software (i.e., REX II).

REX is implemented using LISP on a Symbolics™ 3650. The data interface was developed using a FLAVORS Technology direct memory access system. REX guidance consists of an algorithmic portion, developed in C, and a heuristic portion developed in LISP. The Joshua™ expert system shell has been used to develop a rule-based system to monitor navigation sensors.

11.2 Intelligent System and Functions

REX combines both heuristic and algorithmic processing to:

- Compute and optionally execute "optimal" thrust commands to fly the nominal flight trajectory based on REX guidance calculations
- Monitor and optionally execute Flight Data File (FDF) crew procedures
- Monitor health of onboard guidance, navigation (including sensors), and propellant use

REX contains a numerical guidance controller. REX guidance algorithms can be enabled/disabled at crew discretion. Part of REX automatic guidance is to select the optimum time to pulse as well as the size of the pulse. The optimization algorithm minimizes vehicle relative motion oscillation which results in minimum fuel consumption as well.

REX also includes guidance heuristics to emulate the way crew members fly and a rule-based expert system for monitoring navigation sensors.

11.3 Human-Intelligent System Interaction Functions

REX assists the operator in monitoring the effects of thrust commands. A Relative Motion Window provides trajectory plots of the vehicle with respect to the rendezvous target. These

---

™ Symbolics is a trademark of Symbolics, Inc.
™ Joshua is a trademark of Symbolics, Inc.

plots include predicted positions (at 5 and 10 minutes in future) to assist the operator in visualizing the trajectory.

REX monitors the execution of procedures. A Procedures Window displays a text representation of a procedure and uses color and blinking to identify location within the procedure (i.e., current activity). A timeline is also used to display both activities that have been executed and planned activities. The operator can annotate timeline entries.

REX also assesses the health and status of onboard guidance, navigation (including sensors) and propellant use. Sensor status is presented on a block diagram illustrating data flow from the sensor to the users of the navigation state.

REX was designed for use by the crew onboard the Space Shuttle. It includes the ability to execute procedures and issue commands to the vehicle control systems. An activity log can be created to store all commands issued to the Space Shuttle control system. All log entries include an identification of the source of the command. REX is the only example of identification of source on messages, except the PDRS HCI design concepts.

The operator can selectively control the intelligent system by enabling or disabling portions of the knowledge base (roughly corresponding to procedure blocks).

REX is designed to provided autonomous rendezvous capability with smooth handover to manual operations as needed. To accomplish this, the system incorporates existing manual procedures as system heuristics. Thus, the intelligent system and the crew member fly the same way, using the same approach (although pilots do not always use heuristics consistently). There is a need to vary the heuristics based on situation, however, which REX cannot currently accommodate. REX guidance tries to meet the following goals:

- Fly the correct trajectory
- Minimize fuel usage
- Minimize plume impingement on target

If other goals become important in off-nominal situations, manual guidance is used.

REX can compute the number of thrust pulses required to fly the nominal trajectory through Prox OPS to docking. This computation is only performed when the guidance computation has been manually enabled by selecting the guidance mode button in the Button Window. Guidance modes are:

- ENABLED
  Automatic computation of guidance and corresponding thrust commands; use of these commands is based on Thrust Command mode setting

- DISABLED
  Manual guidance based on feedback of actual position/velocity from sensors and visual cues

REX guidance must be enabled for thrusting commands to be computed. The way that these thrusting commands are used depends upon the Thrust Command mode:

- MONITOR
  Thrusting commands are suggested by REX for manual execution using the translational hand controller (THC); commands are optionally verbalized using speech synthesis[1]

- CONTROL
  Thrusting commands (including desired Digital Auto Pilot (DAP) settings) are issued directly to the vehicle (actually to the crew training simulator in the Space Shuttle Engineering Simulation, or SES) and automatically executed without crew intervention

Interaction between the Guidance mode and the Thrust Command mode results in 3 possible Prox OPS modes:

- Manual docking based on feedback of actual position/velocity from sensors and visual cues

- Manual docking with REX guidance providing suggested thrust commands

- Autonomous docking based on REX guidance

REX can support both real-time operations in the SES and playback operations stand-alone. The selection of data source is made by the operator. Recorded SES data is used to develop a library of test cases for verification and validation as well as for stand-alone operations.

11.4 Supporting User Interface Capabilities

The user interface resembles the format of the Flight Data File, a mission-specific document describing procedures, checklists, and other mission support information. Representations include activities on a timeline, text procedures, and relative motion plots.

The screen is divided into two regions. The left half of the screen contains a fixed window configuration with optional menus for system control. The right half of the screen can be configured by the operator. The right half has been further subdivided into an upper and lower region. See figure 11-1 for the workspace layout.

---

[1] It is envisioned that spoken commands would assist the operator during close proximity operations by providing necessary guidance information while allowing the operator to watch the rendezvous target.

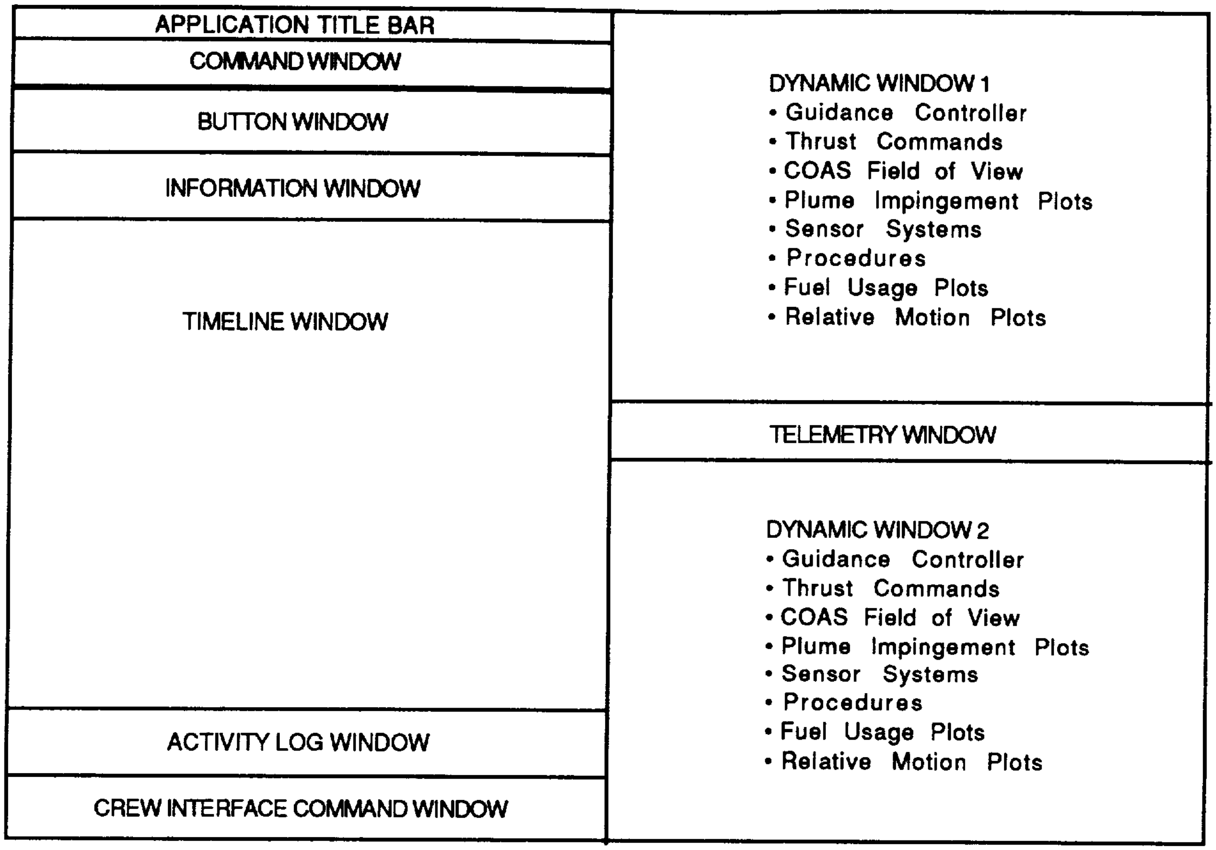


Figure 11-1. Workspace Design of REX II

The normal window configuration shows the relative motion plot in the lower right window. The upper right window is typically used as a dynamic display region, where the other available windows are displayed as needed. Any combination of windows is possible, however.

The remainder of this section consists of a discussion of the information presented in each of these windows.

REX Command Window

This window provides menu options for controlling REX, controlling the display of information on the screen, review of procedures and timelines, and execution of procedures. The following command options are available:

1) Run OPS

   Control of the intelligent system REX is performed from this menu. Control options include starting, pausing, resuming, capturing data from the SES for off-line testing, and quitting the application.

2) Display OPS

   The configuration of the current display is controlled using this option.

3) Procedure OPS
Procedures can be reviewed or executed using this option.

4) Timeline OPS
The Activity Timeline can be reviewed using this menu option.

5) Hardcopy OPS
Either the entire screen or a specific window can be hardcopied from this menu option.

6) Extras
This option provides the operator some control over the application configuration. The operator can enable/disable the Controller, the Sensor Systems intelligent system, or Procedures (i.e., portions of knowledge bases associated with specific procedures). Additionally, access to information for initial synchronization with the simulator is available. Storage of current display configuration via Save Window command is also possible from this menu item.

See figure 11-2 for an example of the REX Command Window.

REX Button Window

1) REX Monitor/Control Mode
The currently selected Thrust Command mode is indicated in the REX Button Window. A thrust command mode button toggles between the two modes at operator selection. The button label reflects the currently selected thrust command mode. CONTROL mode is indicated by the black text message "CONTROL" in the button label on a white background. MONITOR mode is indicated by the text message "MONITOR" in the button label and by a black background surrounding the white text of the mode button label. During both modes the button has a red border.

During MONITOR mode, verbalized thrust commands are possible using speech synthesis capability. The information conveyed in audio is the information on the THC controller display (i.e., number of pulses, direction of pulses). Command enunciation can be suppressed by turning down the synthesizer volume.

2) Status History Buttons
Three status history buttons allow access to the activity history of thrust commands issued from the THC, the DAP, or the Multifunction CRT Display System (MCDS, i.e., keyboard), respectively. The Status History is displayed in a window on the lower left side of the screen. These buttons also indicate the current status of the execution of thrusting commands. These status are:

- THC activity: shows white background when THC commands are issued
- DAP activity: shows white background when DAP panel commands are issued
- MCDS activity: shows white background when a crew keyboard entry is made

These indicators are only active when REX is issuing thrust commands autonomously (REX CONTROL MODE).

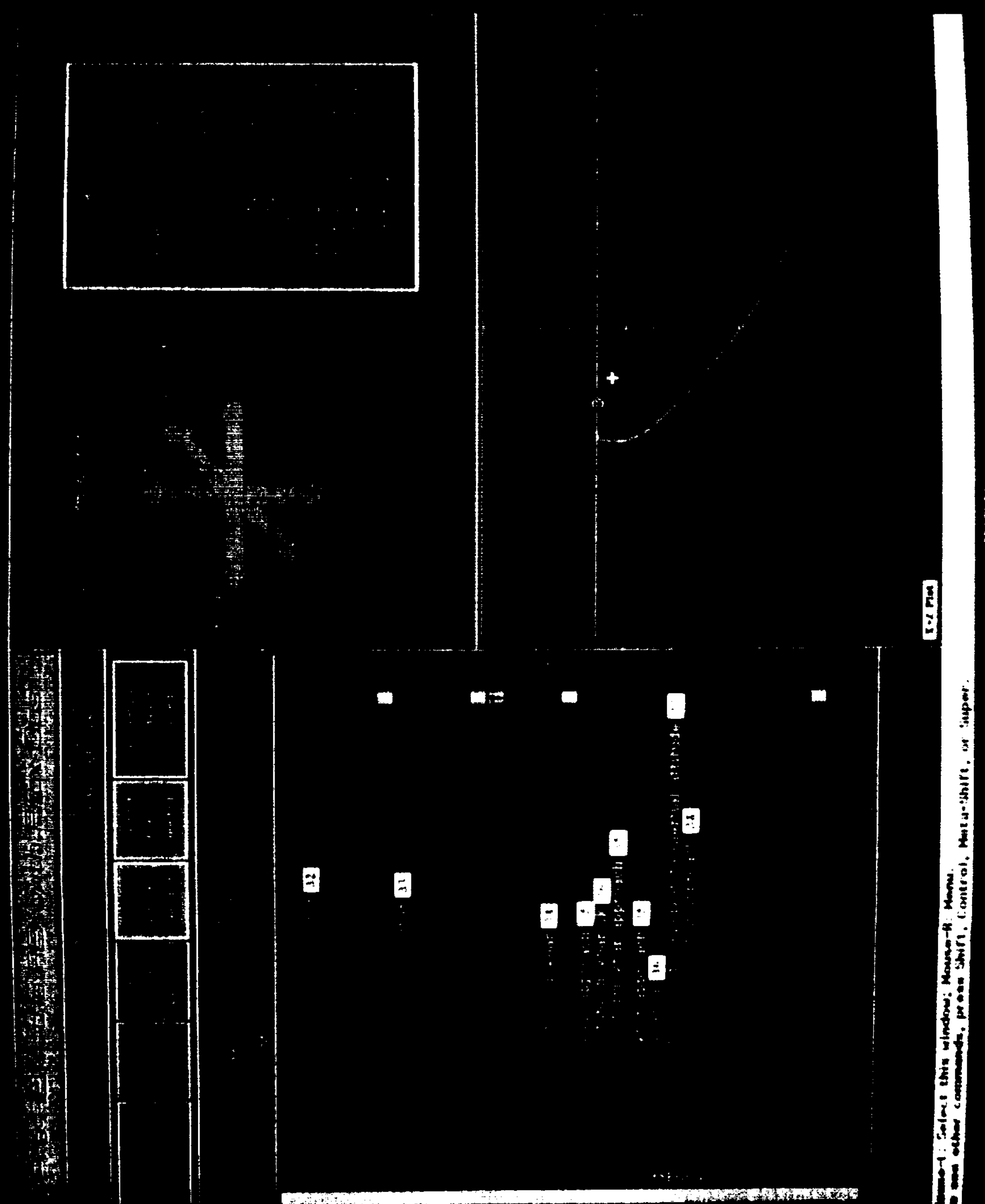


Figure 11-2. Color Example of the REX Display

3) Toggle Display Button
This button allows the operator to switch between the current display configuration and the previous display configuration

4) System Sensors
Selection of this button located in the upper left region of the screen displays the Rendezvous Sensor window in the dynamic workspace. This window is the interface into the intelligent system monitoring navigation configuration that was developed using Joshua. This display consists of a block diagram showing data flow from the selected sensor to the users of the navigation state. Blocks contain a label identifier and an arrow oriented to indicate status (i.e., up-good; horizontal-unavailable or cautionary; down-bad). Blocks are also color coded to indicate status (i.e., green-go; yellow-unknown, unavailable, or cautionary; red-no go). Blocks are connected by lines that change to indicate current data flow. The general flow of data is from a sensor to the navigation software, then into a storage buffer for distribution to the state vector user community. The blocks for navigation software also show statistics about the processing of the data (e.g., number of measurements edited, variation of measurements with respect to expected variance). See the upper right portion of figure 11-3 for an example of the System Sensors Window.

5) REX Guidance Enabled/Disabled
The guidance mode button operates as a toggle switch, so that selection of the button alternates between enabling and disabling REX guidance. Guidance is computed when the button label shows REX GUIDANCE ENABLED in black text displayed on a white background with purple border. Guidance is not computed when the button label shows REX GUIDANCE DISABLED displayed in white text on a black background with purple border.

See figure 11-2 for an example of the REX Button Window.

REX Information Window

1) Dynamic Time Values
The following values of current time are displayed:
- Run-Elapsed Time (RET)
- Mission-Elapsed Time (MET)
- Phase-Elapsed Time (PET)

2) REX Thrust Commands
An alternate display of thrust values to the THC coordinate window is provided in the REX Information Window. This window shows the currently recommended thrusting commands, displayed horizontally from left to right as up/down, left/right, back/in. Only the thrust command for the selected DAP are shown. The current thrust command mode is also indicated in this window. A box surrounds thrust pulse values in CONTROL mode, no box is shown in MONITOR mode.

3) Procedure Activity Needed
The text of this button displays a required procedure or crew activity. The button flashes when the required activity can be executed by button selection.

See figure 11-2 for an example of the REX Information Window.

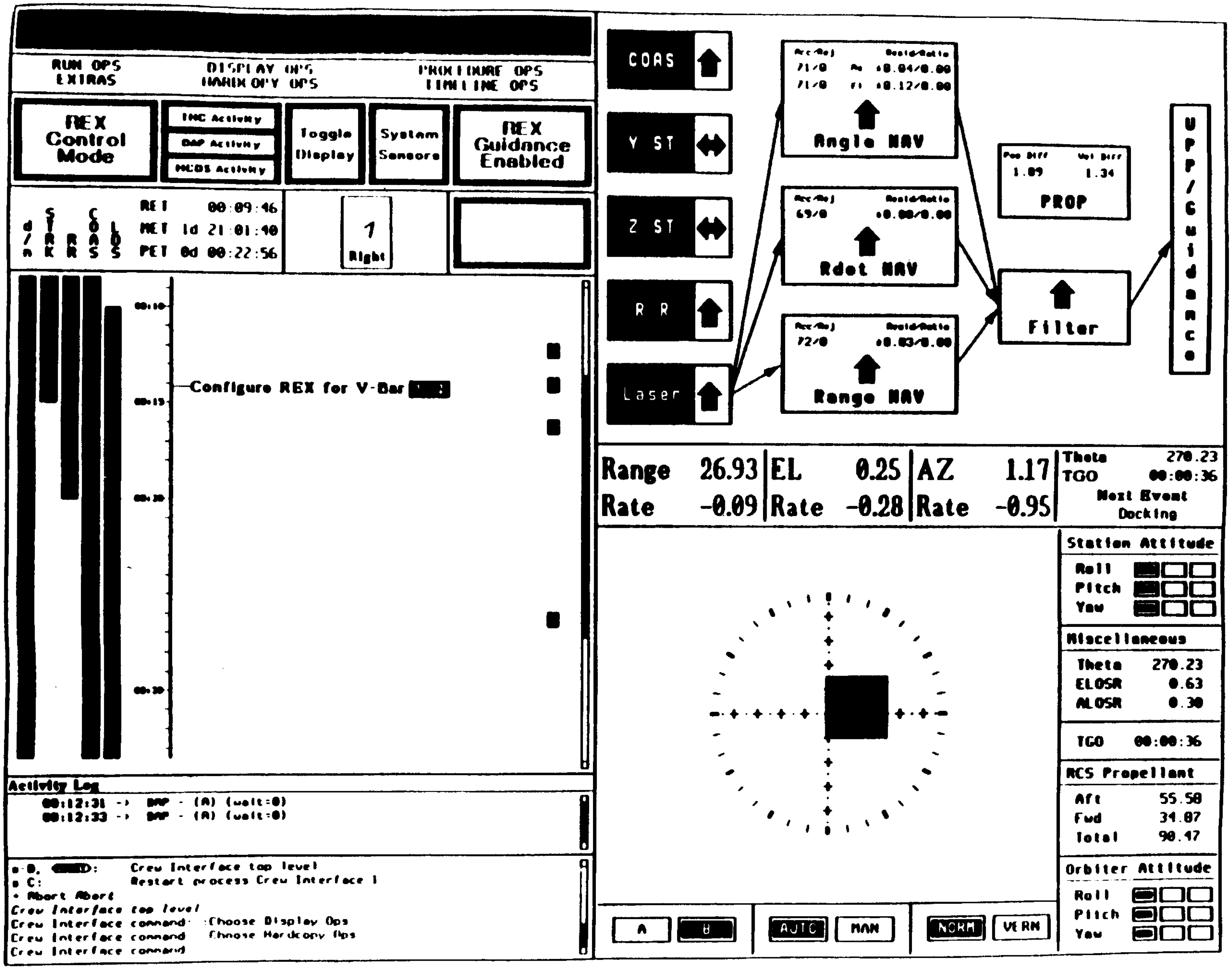


Figure 11-3. Example Illustrating the System Sensors Window and Crew Optical Alignment Sight (COAS) Field of View Window

Timeline Window

The timeline window displays information about the FDF Activity Timeline. Related information includes the current sensor status and availability, the day/night indicator, activities both accomplished and planned, and operator annotation about these activities. The timeline is scrollable and thus represents an event log as well.

The display is organized as a vertical timeline that scrolls off the top of the window. Regions from the left to right of the window are:

1) Day/night Indicator
   Day is white, night is black, and transition periods are marked in grey

2) Sensor Availability
   Available regions are shown as colored bars while unavailable regions are shown as empty spaces; a unique color is assigned to each sensor:
   - Startracker: purple
   - Rendezvous radar: dark blue
   - Crew Optical Alignment Sight (COAS): green
   - Laser docking sensor: red

3) Time Scale
   Time values are displayed in PET, with the current time marked by two opposed triangles

4) Timeline Activities
   Text description of the timeline activity (planned or actual) to the right of the corresponding time value on the time scale. Activities that should be executed are indicated on a flashing yellow background. Activities that have been executed are displayed on a purple background. Activities not yet ready for execution are display in white text.

5) Operator Annotations
   Two types of annotations are provided:
   - Notes may be added to the timeline, either associated with an existing timeline activity or at a specific time
   - An emulation of yellow stick-on notepads can be used to associate an annotation with a specific region of the timeline

Control options within the Timeline window:

1) FDF Timeline Options
   Mouse selection of a blank region displays a menu for adding operator annotations, either Notes or Yellow Stick-on Notepads.

2) FDF Timeline Activity Options
   Mouse selection of an activity label displays a menu providing control options for the selected procedure. Options include review of the procedure text or execution of the procedure.

3) Contents of Operator Annotations
   The contents of a Note or Yellow Sticky can be reviewed, edited, or annunciated with the speech synthesizer by mouse selection of the icon representing the annotation.

4) Scroll Bar
Mouse selection of the scroll bar alters the time region displayed in the window.

See figure 11-2 for an example of the Timeline Window.

REX Activity Log

The REX Activity Log keeps a history of all REX commands sent to the SES. This log is displayed beneath the timeline window. It is displayed as a scrollable, time-sorted message list. A message consists of a timetag, the source of the command, the command arguments, and the elapsed time until the command will be processed. This log can be saved to file by double-clicking the middle mouse button in the log window. See figure 11-3 for an example of the REX Activity Log.

Crew Interface Command Window

This window displays the MCDS or crew keyboard entries. This is a scrollable window and thus provides a review capability for crew entries. See figure 11-3 for an example of the REX Crew Interface Command Window.

Telemetry Window

The central section of the right portion of the screen always displays raw sensor measurements from the selected rendezvous sensor (range, azimuth, elevation). This region also displays the next expected event and the time until that event is expected to occur (i.e., time to go (TGO)). Nominal events are recognized by the system. The definition of which events would be nominal is based upon the options selected by the operator.

Guidance Controller Options

The operator can customize the type of guidance computed. Three guidance profiles are available: transition (through 800 feet), final approach (through docking), and stationkeeping. Guidance control options are entered from the right portion of the screen when it has been configured to display the Guidance Controller Options (see Command Window). Options include:

1) Direction of Final Approach
   - VBAR - downtrack from target
   - -RBAR - from above target
   - +RBAR - from below target

2) Automated Final Approach
   - On - auto capture of Line of Approach (LOA)
   - Off - manual capture of LOA

3) Braking gates
   - Transition braking gates - on/off
   - Final approach braking gates - on/off

4) Selection of state vector for guidance computation
   - NAV - state computed by onboard flight software
   - Environment - state provided by simulator

5) Docking guidance
Attitude error correction at docking - on/off

6) RDOT control - type of control for VBAR approaches; RDOT refers to range rate
- Bleed - no rate control, except minimum 0.1 feet per second (FPS)
- Constant - constant rate control, value selected by crew

7) Torque equilibrium angle approach
- 5 degrees to +5 degrees

8) Burn times
- T1 defaults to 5 minutes RET with 1 minute delays
- T2 is selected via menu input in MONITOR mode; the optimized solution is computed automatically by REX in CONTROL mode

9) Target coordinates for each approach axis (downtrack-VBAR, crosstrack-HBAR, radial-RBAR)
- Transition
- Final approach

The method of entering guidance control options includes mouse selection of desired value from the range of all possible values and keyboard entry of values. See the lower right portion of figure 11-4 for an example of the Guidance Controller Options Window.

Thrust Command Window

A collection of windows associated with thrust commanding can be displayed in the dynamic workspace. In the upper left hand portion of the window, the actual thrust values are shown using orthogonal axes (THC coordinates) to indicate pulse direction and numeric values positioned near the axes to specify number of thrust pulses. Pulses are displayed in sets, where the first value specifies the number of pulses with a coarse DAP setting (A) and the second the number of pulses with a fine DAP setting (B). The value corresponding to the current DAP setting is displayed in red and the other value is displayed in white. The axes are also color-coded, where blue is used for -Z sense switch setting and yellow for the -X sense switch setting.

The state of the DAP is displayed beneath the THC display in the dynamic workspace. The state of the DAP consists of a setting (i.e., A for coarse or large pulses and B for fine or small pulses), a mode (i.e., automatic or manual) and a thruster specification (i.e., normal or vernier). The current values are indicated in white text on a blue background.

To the right of the THC display is a vertical alignment of windows displaying information related to thrusting:

1) Station Attitude
The status of the current orientation of the rendezvous target (in this case the Space Station) with respect to the specified orientation is shown in this window. The allowed angular variation from the expected orientation is the deadband. Three panel lights are shown for each rotation axis (i.e., roll, pitch, yaw). The left-most panel shows green if the vehicle is within deadband on that axis. The middle panel shows yellow if the vehicle is between 1 and 2 deadbands on that axis. The right panel shows red if the vehicle exceeds 2 deadbands on that axis.

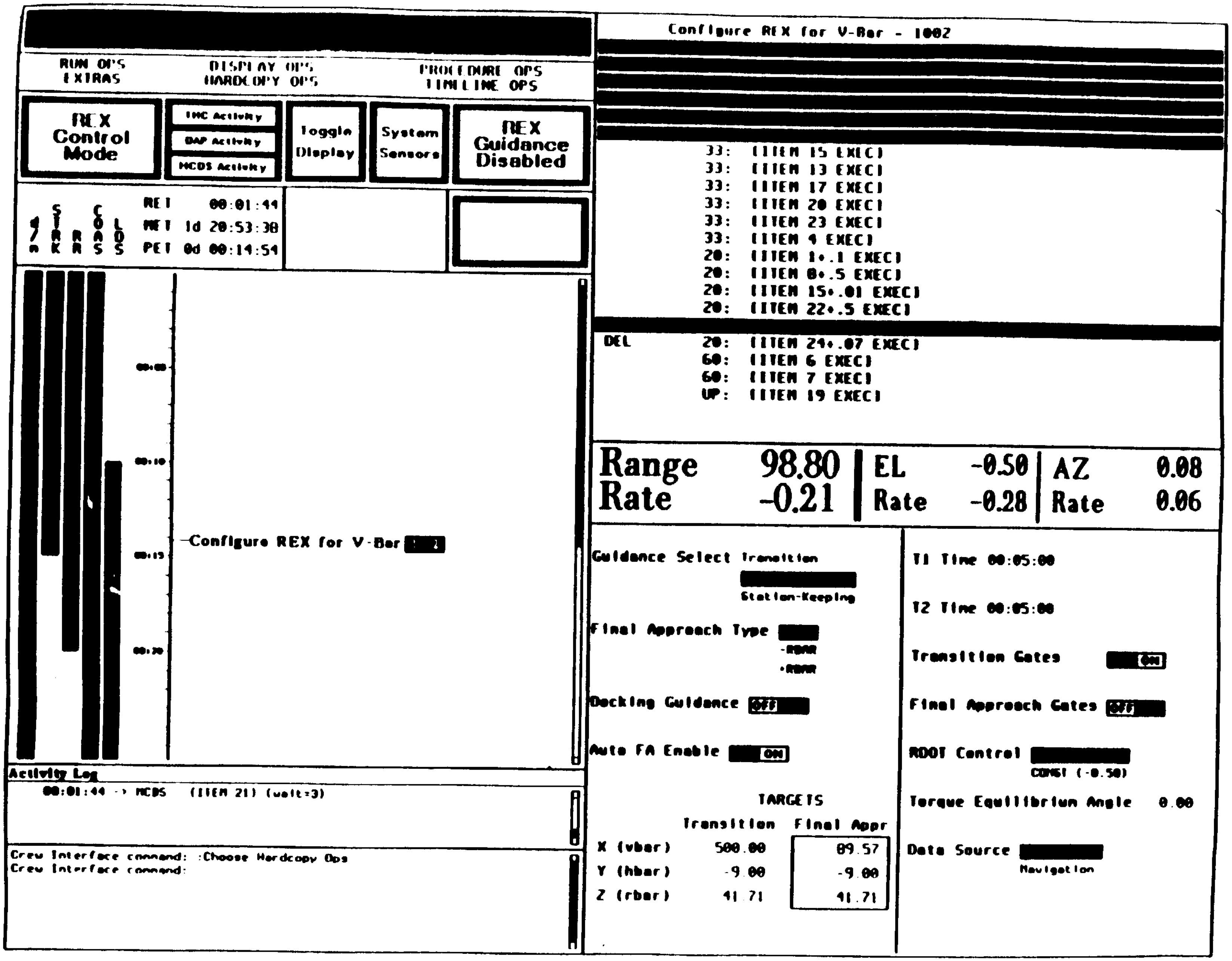


Figure 11-4. Example Illustrating the Guidance Controller Options and the Procedures Window

2) Miscellaneous

3) Time to Go (TGO)
   Prior to a burn, the predicted burn length; during a burn, the expected time until burn completion

4) RCS Propellant
   Propellant reserves in the forward and aft tanks, as well as the total reserve

5) Orbiter Attitude
   The status of the current orientation of the Space Shuttle with respect to the specified orientation is shown in this window. The color coding is the same as that for "Station Attitude".

See upper right portion of figure 11-5 for an example of the Thrust Command Window.

COAS Field of View Window

The COAS field of view out the selected window can be optionally displayed instead of the THC coordinate system. This window shows the alignment grid overlaid on a graphic of the target to illustrate REX's interpretation of the current vehicle position with respect to the environment (i.e., real world). As the vehicle approaches the target, the target graphic gets larger. See lower right portion of figure 11-3 for an example of the COAS Field of View Window.

Procedures Window

Text of a procedure can be shown in the dynamic workspace, upper right. The procedure execution can also be monitored using this window. Steps of the procedure that have been executed are shown on a purple background. The next step expected to be executed blinks and is shown on a green background. The current prototype contains a subset of the procedures used during Space Shuttle rendezvous and Prox OPS. See the upper right portion of figure 11-4 for an example of the Procedures Window.

Relative Motion Window

The relative motion plots display the actual position of the vehicle with respect to the target and two predicted positions. At ranges exceeding 1000 feet, predicted positions at 5 and 10 minutes are displayed. At ranges under 1000 feet, the interval between predicted positions drops to 1 and 2 minutes, respectively. The next predicted position is shown as a circle and the second predicted position is shown as a plus sign. Trajectory predictors can be manually disabled from the interface.

The following viewing options are available for the Relative Motion Window:

1) Mouse-scaling
   Zoom in for closer inspection or zoom out for an overview

2) Braking gates
   Determine when the range gates have been entered or exited (uses "Saturn-ring" approach)

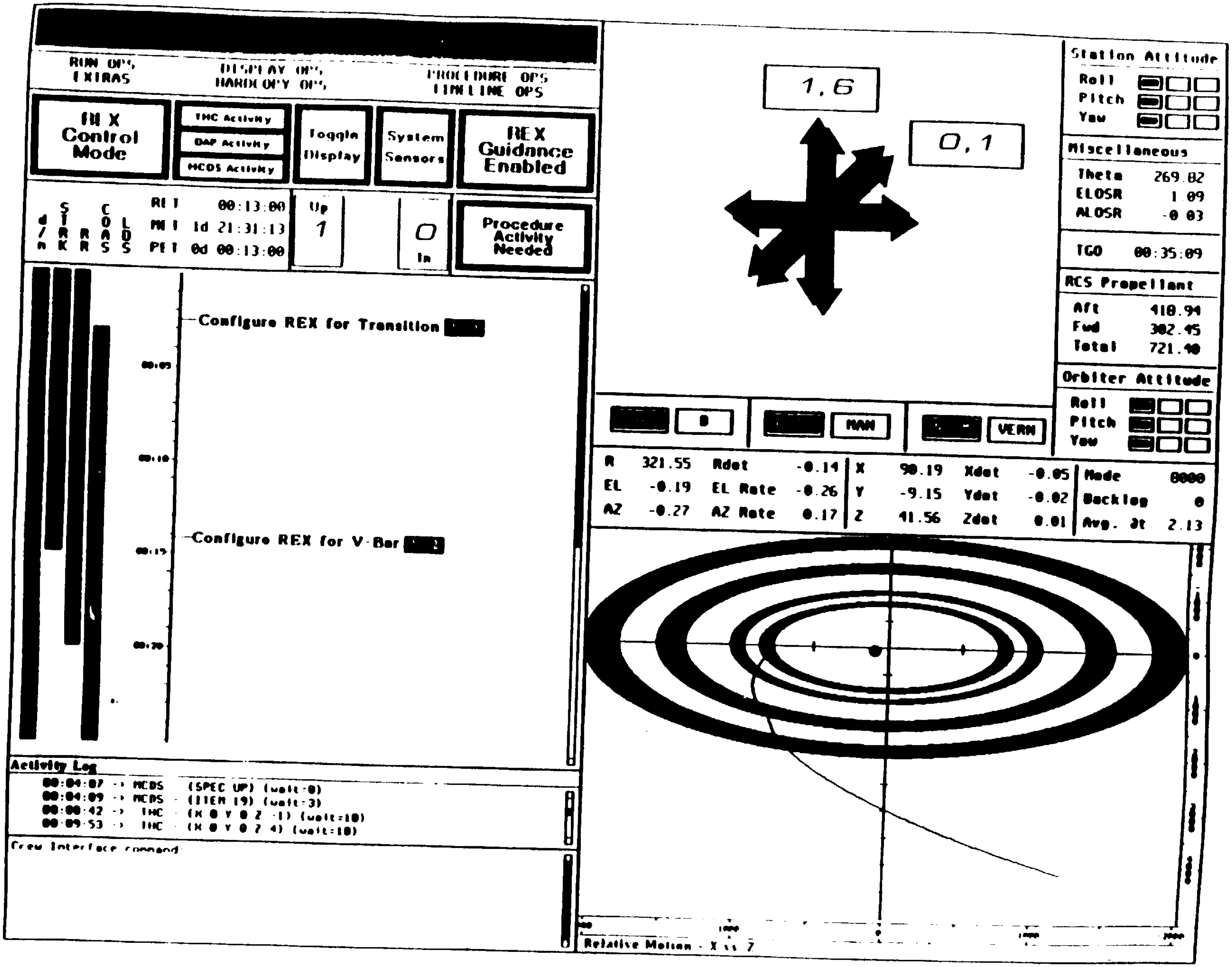


Figure 11-5. Example Illustrating Thrust Command Window

3) Pulse predictor
   Predict and display the trajectory effects of immediate execution of two translational thrust pulses (large pulses using DAP A)

4) In-plane/out-of-plane views
   Superimpose the out-of-plane relative motion plot on the in-plane view; color and scale are used to distinguish the plots

See the lower right portion of figure 11-2 for an example of the Relative Motion Window.

Plume Impingement Plot Window

Plume impingement plots show the force on the target due to vehicle thrusting as a function of distance to the target. Actual plume impingement data is compared to the pre-computed average plume impingement. See the upper right portion of figure 11-6 for an example of the Plume Impingement Plot Window.

Fuel Usage Plot Window

A plot of fuel usage is also available, but was not viewed by the study team.

11.5 Design Process

REX was developed in two phases, paralleling the two phases of a Space Shuttle rendezvous with a target orbiter (i.e., Rendezvous and Prox OPS). The first phase of REX was designed for the rendezvous phase. REX I was enhanced to include the Prox OPS region. REX II included a complete redesign of the user interface to accommodate Prox OPS.

The initial requirements were based on domain expertise resident in the development group and on related documentation. These requirements were documented in a formal requirements document (Olszewski, 1989). Requirements were reviewed by members of both the Space Shuttle crew and ground flight operations personnel. The review process has continued through out iterative development by collecting feedback from demonstrations.

At the time of the interviews, "internal" testing and certification (i.e., testing within the development organization) had been performed, although user comments have been elicited through review meeting and demonstrations. Test cases recorded in SES are being used to validate the system. REX has been integrated with the SES for testing in an operations-like environment.

User interface design included human factors considerations on the role of color in the system. The importance of distinguishing the user interface from the developer interface was also recognized. The design of user interface was driven by user needs. They have employed user review through demonstration to upgrade the user interface.

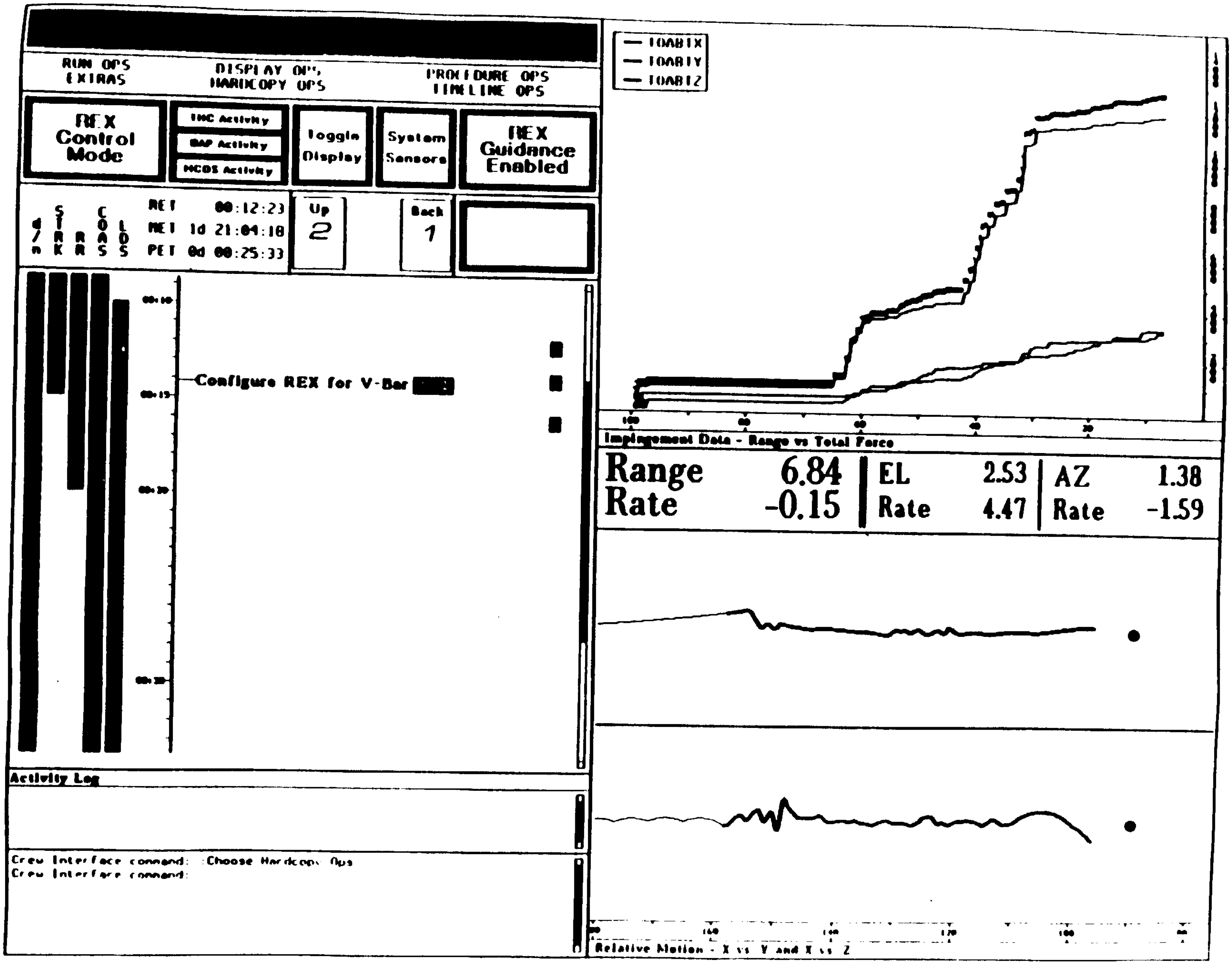


Figure 11-6. Example Illustrating the Plume Impingement Plot Window

11.6 Study Method

Information about REX was obtained by interview of the project representatives, demonstration of the prototype, and review of the case data sources cited below. The first demonstration in January of 1990 was conducted stand-alone in the AI lab of the Intelligent Systems Branch at JSC. In August of the same year, a second demonstration of the system was observed. In the second demonstration, REX II was integrated with the SES.

Study Team

- Jane Malin (NASA Johnson Space Center)
- Debra Schreckenghost (The MITRE Corporation)

Project Representative

- H.K. Hiers (NASA Johnson Space Center)
- Oscar Olszewski (Lockheed Engineering and Science Company)

11.7 Case Data Sources

Hiers, H.K., (July, 1990), "Rendezvous/Prox OPS Expert System (REX) for Crew Procedures Demonstration", briefing charts for demonstration, Johnson Space Center, Houston, TX: NASA.

Olszewski, Oscar (May, 1989), *Rendezvous Expert System Requirements Definition Document*, LESC-26542, Houston, TX: Lockheed Engineering and Sciences Company.

Olszewski, Oscar (June, 1989), "Rendezvous Expert System (REX2) Requirements Review User Issues", briefing charts, Houston, TX: Lockheed Engineering and Sciences Company.

Section 12
Operations Management System (OMS) Prototypes

12.1 System Description

The Operations Management System (OMS) will coordinate nominal operations and manage faults for the major Space Station Freedom Program (SSFP) systems. The OMS includes human actions as well as software, both onboard the Space Station and on the ground. A set of advanced automation prototypes were developed to demonstrate the subset of the OMS responsible for failure management, specifically integrated global fault detection, isolation, and recovery capabilities for SSFP systems. Three of these OMS prototypes were evaluated, the Diagnostic Reasoner (DR), the Recovery Expert (Rx), and the Procedures Interpreter (PI). The Diagnostic Reasoner diagnoses failures based on input from the major Space Station systems. The Recovery Expert provides planning for recovery based on the diagnosis from the Diagnostic Reasoner. The Procedures Interpreter monitors and executes the procedures identified by the Recovery Expert. For all of these prototypes, the human remains a key element in fault management. Humans are required to make final decisions and to handle unexpected situations. Figure 12-1 provides an overview of the OMS Prototypes that were evaluated.

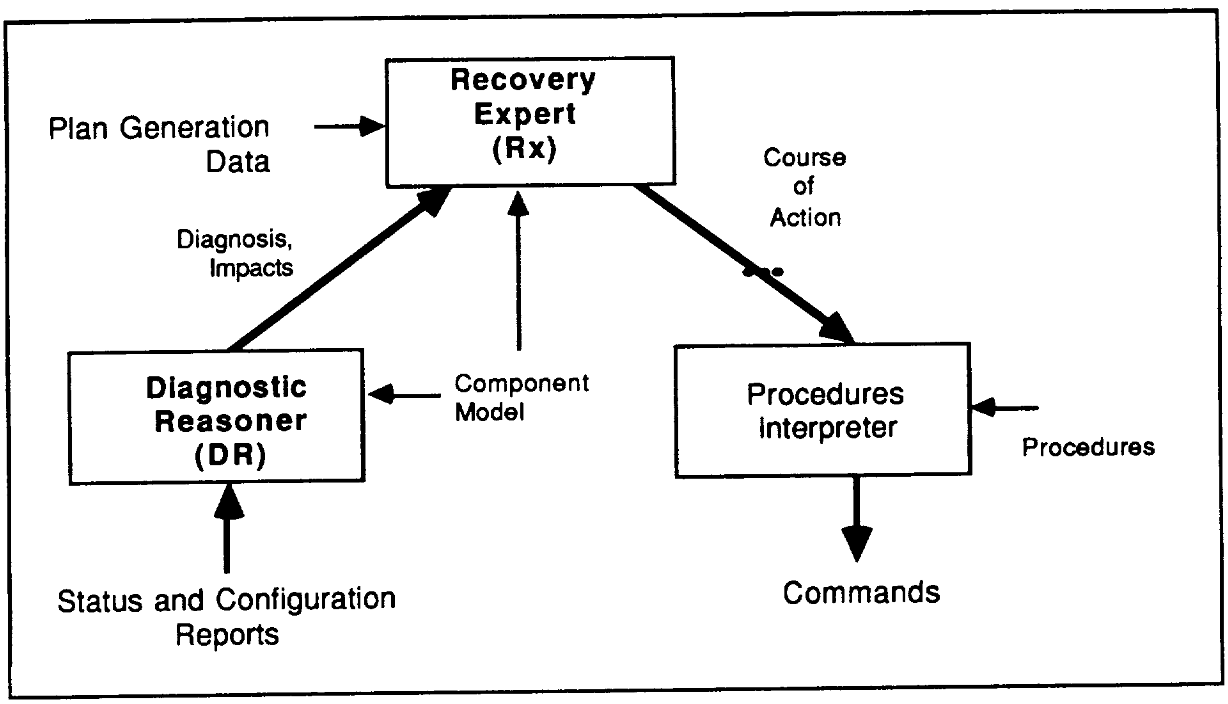


Figure 12-1. Overview of the OMS Prototypes (Baker et al., 1991)

The OMS Prototypes were derived from concepts for integrated management of hierarchical distributed systems that were developed for SSFP in the mid 1980's. Initially, two prototypes were implemented for the Mission Operations Directorate at JSC, the Integrated Status Assessment (ISA) and the Procedures Interpreter (PI). The ISA diagnosed the cause of a failure and PI executed and monitored recovery procedures resulting from the failure. These systems were developed stand-alone in Zetalisp® on a Symbolics™. These prototypes were later integrated into the Data Management System (DMS) Test Bed at JSC to provide OMS capability for the End-to-end Test Capability (ETC) demonstrations. Eventually, ISA and PI were ported to a more standard set of hardware and software tools. Specifically, PI was ported to a VAX® workstation and the Operations and Science Instrument Support (OASIS) Teleoperations Software Package that is written in Ada®. ISA was ported to C and C Language Integrated Production System (CLIPS) on an IBM PC®/AT. Using this configuration, a number of ETC demonstrations were supported. See Schreckenghost and Kelly, 1989, for a detailed description of the early OMS Prototypes.

The current OMS prototyping effort is funded jointly by NASA's Engineering Directorate and MITRE. These prototypes are based on the earlier OMS prototypes (i.e., ISA and PI) and are also planned for use in the DMS Test Bed. The Diagnostic Reasoner is a derivative of the ISA with enhanced capability, including failure impact assessment. The Recovery Expert is a new module that provides goal-directed planning capabilities. The VAX version of PI will be integrated with these applications to provide procedures monitoring and execution capability. The Diagnostic Reasoner and Recovery Expert were developed on a VAX workstation using Ada ART™ (with LISP interpreted to Ada at run-time), Ada, and the user interface package Transportable Application Environment (TAE™). In later versions, the user interface will be ported to the X Window System™ and the application software to Ada.

12.2 Intelligent System and Functions

Diagnostic Reasoner

The Diagnostic Reasoner (DR) is a model-based system that generates a set of suspected faults that would result in the observed fault symptoms, predict the impacts of these faults, and, where possible, exactly identify the fault (or *isolate* the fault). Input to DR consists of reports from the major Space Station systems (Tier 2 systems). These reports are used to update the component model. If anomalies are detected, a failure suspect list is generated. An impact sequence that project failure effects forward in time can also be generated if needed. Both the suspect list and impact sequence are passed to Rx for planning anomaly response.

DR models two aspects of Space Station components: structural and functional. Models of behavior in the presence of faults are available. Using these models, a list of suspected faults is generated. One suspect list is generated for every suspected problem and each list has a unique identifier. A suspect list includes the following information:

- Suspects
  Components suspected as the cause of a failure
- Likelihood[1]:

® Zetalisp is a registered trademark of Symbolics, Inc.
™ Symbolics is a trademark of Symbolics, Inc.
® VAX is a registered trademark of Digital Equipment Corporation
® Ada is a registered trademark of the United States Department of Defense
® IBM PC is a registered trademark of International Business Machines Incorporated
™ Art is a trademark of Inference Corporation
™ TAE is a trademark of NASA
™ The X Window System is a trademark of MIT
[1] Data believability and fault likelihood have not yet been incorporated into the system, but are planned enhancements during 1991.

Statistical assessment of the likelihood that a suspected fault is the cause

- Believability[1]:
  Heuristic assessment of the reliability of data supporting the conclusion that the suspect is cause

- Impact sequence:
  Expected impacts resulting from the failure of the suspected component

- Impact severity:
  Assessment of the severity of these impacts[1]

- Explained behavior measures:
  List of the measured behaviors that are explained by the suspect

- Unknown behavior measures:
  List of needed behavior measures to determine if suspect is the cause

The impact sequence is computed using an impact model that captures cause and effect relationships. The following information is modeled for a failure at a given location:

- Effect of the failure
- Severity of the failure
- Delay before the impact is manifested
- Conditions that must exist for the failure impact to occur

Recovery Expert

Recovery Expert (Rx) is a goal-directed planning application used to determine a Course of Action (COA) to recover from a diagnosed failure. Rx uses the results from DR as a starting point in planning recovery activities, specifically the suspect list and the failure modes and effects criticality analysis (i.e., failure impact assessment). Planning goals include isolation of a failure, mitigation of the impacts of a failure, or recovery from a failure. A goal-activity network is constructed to accomplish the specified goal. A baseline assumption is that goals are independent. Activities have entry conditions (pre-existing situation), pre-requisites (activities that must precede the activity), exit conditions (post-activity situation). Post-requisites (activities that must follow the activity) are a planned enhancement (e.g., after a repair, restore the system to nominal working conditions). Entry conditions can't be changed by planning while pre-requisites can. The goal-activity network is expanded until all pre-requisites are satisfied or conflicts in goal states or activity entry conditions are detected. Part of the evaluation of this network is the identification of homologues (i.e., likeness in structure). Homologues should be differentiated from analogues (i.e., likeness in function but difference in structure). Identification of analogues is the basis of Case-Based Reasoning (CBR).

---

[1] Impact severity was hard-coded at the time of the interview, but an upgrade to heuristic assessment is planned.

The goal-activity network is traversed to identify a set of possible COAs. All COAs are evaluated based on three key parameters:

- Schedulability
- Timing
- Undesirability

These parameters are combined into a score used by the operator to evaluate the optimality of the COA. The larger the score, the more optimal the COA (with respect to the parameters). The algorithm used to combine parameters is taken from a decision support method, the Technique for Rapid Impact Analysis and Goal Evaluation (TRIAGE) developed by Decision Science Applications (DSA) Inc (DSA, September 1990). Note that a logarithmic time scale is used (e.g., 0 for 1-10 second, 1 for 10-100 second, 2 for 100 - 1000 second, etc). See figure 12-2 for an illustration of the logarithmic representation of relative timing.

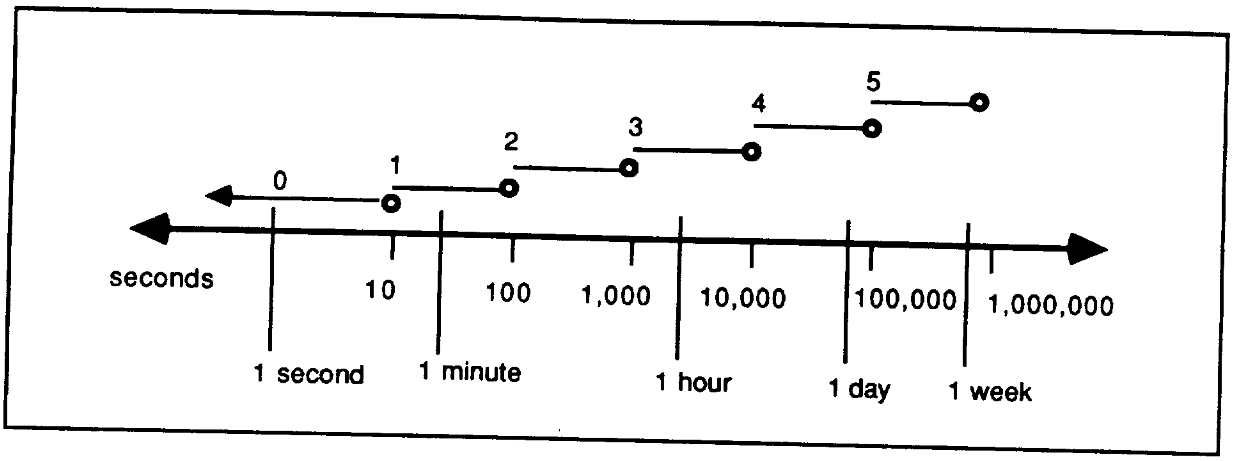


Figure 12-2. Logarithmic Representation of Relative Timing (Baker et al., 1991)

Rx generates multiple COAs, based on partial COAs that address specific aspects of a problem. This set of COAs is then evaluated by the operator, who selects the COA ultimately used. Three aspects of a problem that can be considered are:

- Equipment damage
- Equipment malfunction
- Resource over-utilization

Different approaches to failure management are considered during planning, such as initiating repair, mitigating impacts, or providing for further diagnosis.

Procedures Interpreter

The Procedures Interpreter (PI) monitors and executes activity sequences that have been pre-defined as procedures. PI also detects anomalies in procedure execution and provides execution status to the operator. Although PI was developed as an independent application, it is planned to integrate PI with Rx and DR, to allow execution of COAs generated when a failure is diagnosed.

There are two versions of PI, the original prototype developed on a Symbolics using Zeta Lisp and a later version of the prototype developed in OASIS[1] on a VAX workstation. The description provided in this report refers to the original prototype on the Symbolics. This version was reviewed since it provided a better implementation of the procedure displays at the time of the case study than the VAX version. Eventually, DR and Rx will be integrated with the VAX version of PI.

PI is subject to constraints in both information and capability. The system does not have knowledge of sensor failures or have access to resource availability. PI executes activities sequentially and cannot simultaneously execute procedures.

12.3 Human-Intelligent System Interaction Functions

Diagnostic Reasoner

DR assesses system status and diagnoses system failures. Diagrams illustrating both structural and functional models are used to present this status information. A schematic illustrating connectivity (i.e., power, data, temperature control) between systems is also provided. DR identifies a list of suspected failures, evaluates how well failures match current conditions, assesses the future impacts of failures if they are not corrected, and identifies information needed to confirm the failure. This information is used by the operator to assess current health of Space Station systems and to determine the potential for problems in the future. This information is also passed to Rx for use in planning for failure recovery.

The operator can optionally display the suspected failures, accompanied by a schematic of the affected system. The intelligent system performs a variety of evaluations that can be used by the operator in assessing this list of suspected failures. These evaluations include (1) the likelihood that suspect is the cause, (2) the reliability of the data supporting failure, (3) a failure impact assessment, (4) a description of behavior explained by failure, and (5) the information required to confirm the suspected failure. Most of this information is presented as text in the message list. Failure impact sequence is displayed using a tree structure, however, with the impacts partitioned into time regions. Messages are displayed in the message list for all events and operator activities. Diagnostic events may also be viewed using a timeline display, where events are grouped by component.

DR is a diagnostic application and provides no capability to intervene with or control the monitored process. The operator can initiate execution of Rx from DR, however.

Recovery Expert

There are two major activities required to generate a plan to recover from a failure. The operator is involved in both of these activities. First, the goal activity network is generated. The operator initiates generation of the network and controls how this network is generated. He can choose to generate the network step-wise, to automatically generate a complete path, or to defer a path. The goal-activity network is illustrated as a graphic of goal icons and activity icons with connectors that indicate the type of relationship and with a related text description in the message list.

Next , the operator initiates generation of a set of COAs based on the goal-activity network. Each COA is illustrated in two ways, as a text description of planned activities and a diagram

---

[1] OASIS is a government-owned off the shelf software tool developed in Ada by the Laboratory for Atmospheric and Space Physics at the University of Colorado.

illustrating the procedural sequence required to execute the COA. Each COA is evaluated using a set of optimality criteria (i.e., schedulability, timing, and undesirability). The operator uses this optimality score for a COA to select between available COAs. The operator can optionally access information about how the COA scored with respect to each of these criteria. The operator can also annotate a particular COA.

At the time of the interview, Rx was not integrated with PI, the application that monitors and executes procedures that make up a COA. This integration was planned for a later version, however.

Procedures Interpreter

PI executes procedures, monitors procedure execution, and detects anomalies in procedure execution. When an anomaly is detected, an alarm is issued. Alarms are displayed as text messages and are used to annotate the procedure step being executed when the alarm occurred. Additional information provided to the operator include a status assessment of the active procedure and relevant data plots.

At the time of the interview, the operator selected the procedure to be executed. When integrated with Rx, it is possible that procedures will be automatically selected based on COA. When a procedure has been selected, the entire procedure is displayed as a sequence text activities. The intelligent system marks which steps in this sequence have been executed and annotates these steps with the status of the execution.

PI has the ability to execute procedures and thus can intervene with or control the monitored process. The operator initiates procedure execution. He also selects from the five modes available for executing procedures (i.e., levels of automation for procedures execution):

- Hardware Switch Manual
  Execution of procedure by operator

- Software Switch Manual
  Execution of procedure by intelligent system at operator request

- Automatic with Confirmation
  Step-wise execution of procedure with operator confirmation at each step; if hardware switch, operator executes procedure; if software switch, intelligent system executes procedure

- Full Auto
  Execution of procedure by intelligent system with no operator intervention required

- "Start-Stop"
  Execution of procedure as designated, where each execution step is pre-specified as one of the above types

Notice how the responsibilities of the intelligent system and the operator vary for each of these modes. Partial execution of procedures is possible, including execution to a break point or buffering a procedure for execution until a specified time.

12.4 Supporting User Interface Capabilities

Diagnostic Reasoner

The workspace for DR consists of five regions:

- Space Station Structure
  Representation of the structural arrangement of the Space Station's physical elements

- Space Station Systems
  Representation of the major systems of the Space Station

- DR Control Panel
  Button panel providing control options for DR. Note that sensitive regions within the other windows provide additional capability.

- Dynamic Region
  This is the region where new information pops up. The default display in this window is an overview schematic of the major systems.

- Message panel
  Scrollable region for displaying messages. This region is actually the size of the full screen, but is buried beneath the other windows such that only the lower portion of this window is visible. It can be brought to the surface if needed.

Figure 12-3 shows the DR workspace.

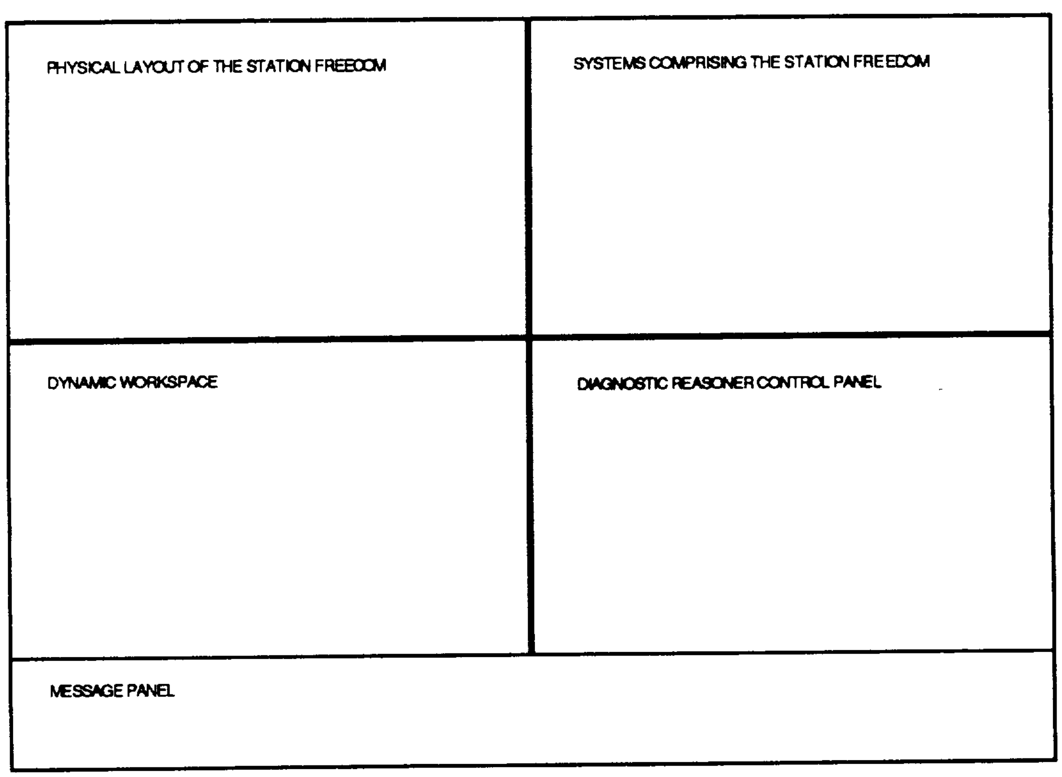


Figure 12-3. Workspace of Diagnostic Reasoner

Both the Structure and Systems graphics represent the highest level of abstraction. The physical elements and systems icons are mouse-sensitive. Selection of a physical element pops up a window in the dynamic region that illustrates the physical location of all components of that element. Selection of a system pops up a window in the dynamic region showing a schematic of the system. The same background shading or color is used for both the high level graphic and the detailed popup to associate the popup with related display regions. Connections between component are color coded to indicate temperature of fluid flowing through the connection (red - warm, blue - cool). Selection of a component displays an assessment of the status of the component in the message panel (e.g., cold plate 15 is nominal).

The default screen in the dynamic region is an overview schematic of the major systems of the Space Station. This schematic includes connectivity between systems for information access, power, and temperature control. As additional windows are called up, they are layered in the dynamic region (i.e., a new popup is positioned on top of previous windows). A window must be exited to be destroyed. Windows are arranged in tiles as a default, but the windows can be moved to overlapping positions if the user desires.

Figure 12-4 shows the top level interface as it appears at initiation of the system.

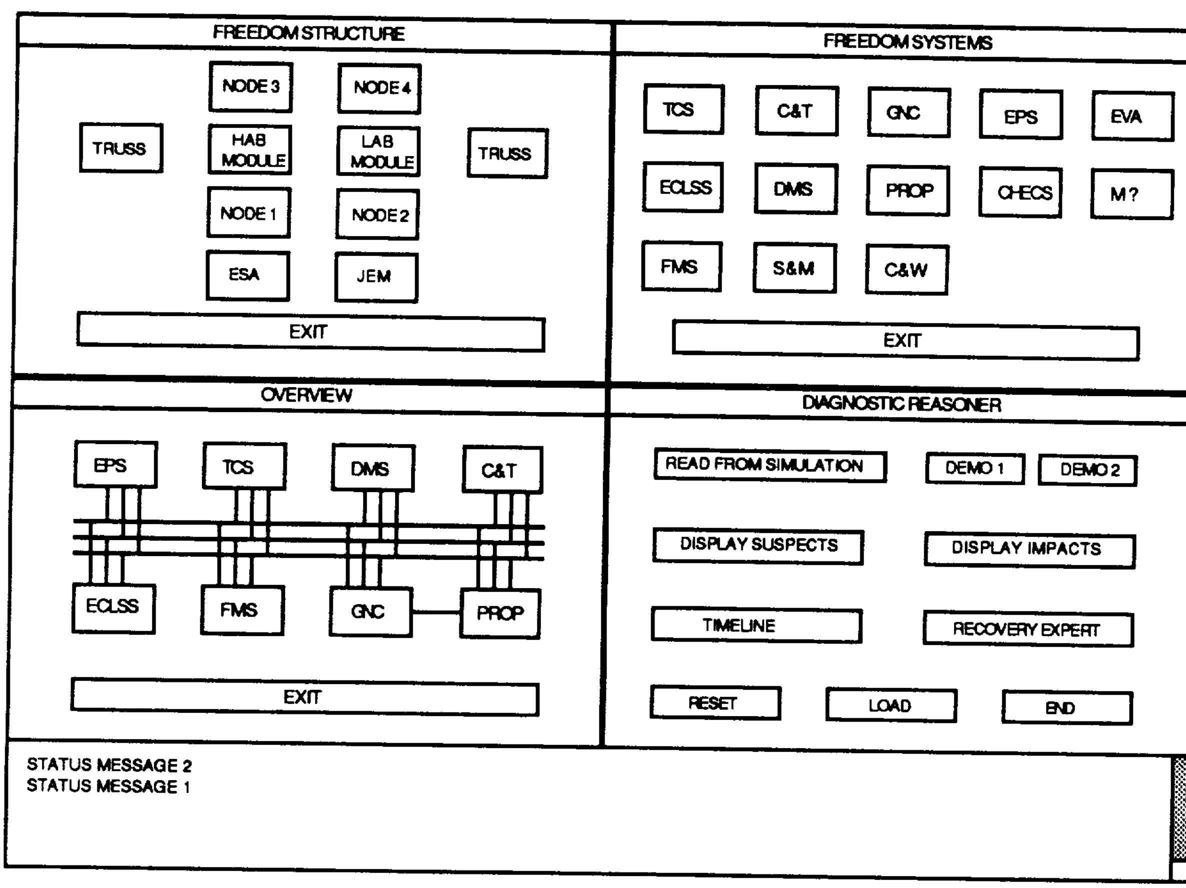


Figure 12-4. Display at Initiation of the Diagnostic Reasoner

The Control panel provides capability to execute the system, to display a suspect list, to display fault impact sequence, to display a timeline of events, and to initiate execution of Rx. Intelligent system control functions include reset of the intelligent system knowledge base, load the knowledge base, and exit the intelligent system.

Currently the intelligent system executes using simulated data based on behavioral models that have been stored in a file. Two data sets are provided. A data set is opened and execution of the intelligent system is initiated by selecting either the button "Demo 1" or "Demo 2". Each time the "Read from Simulation" button is selected, another time slice of data is processed from the file.

Selection of the "Display Suspects" button displays the suspect list in the message panel (see figure 12-5). Selection of the "Display Impacts" button displays the failure propagation potential, a hierarchy of the possible sequences of fault impacts propagated into the future on a logarithmic time scale (see figure 12-6). Notice that this display can be large and overwrite more than the dynamic region. An interesting aspect of this display is the designation of time regimes within a hierarchical tree structure.

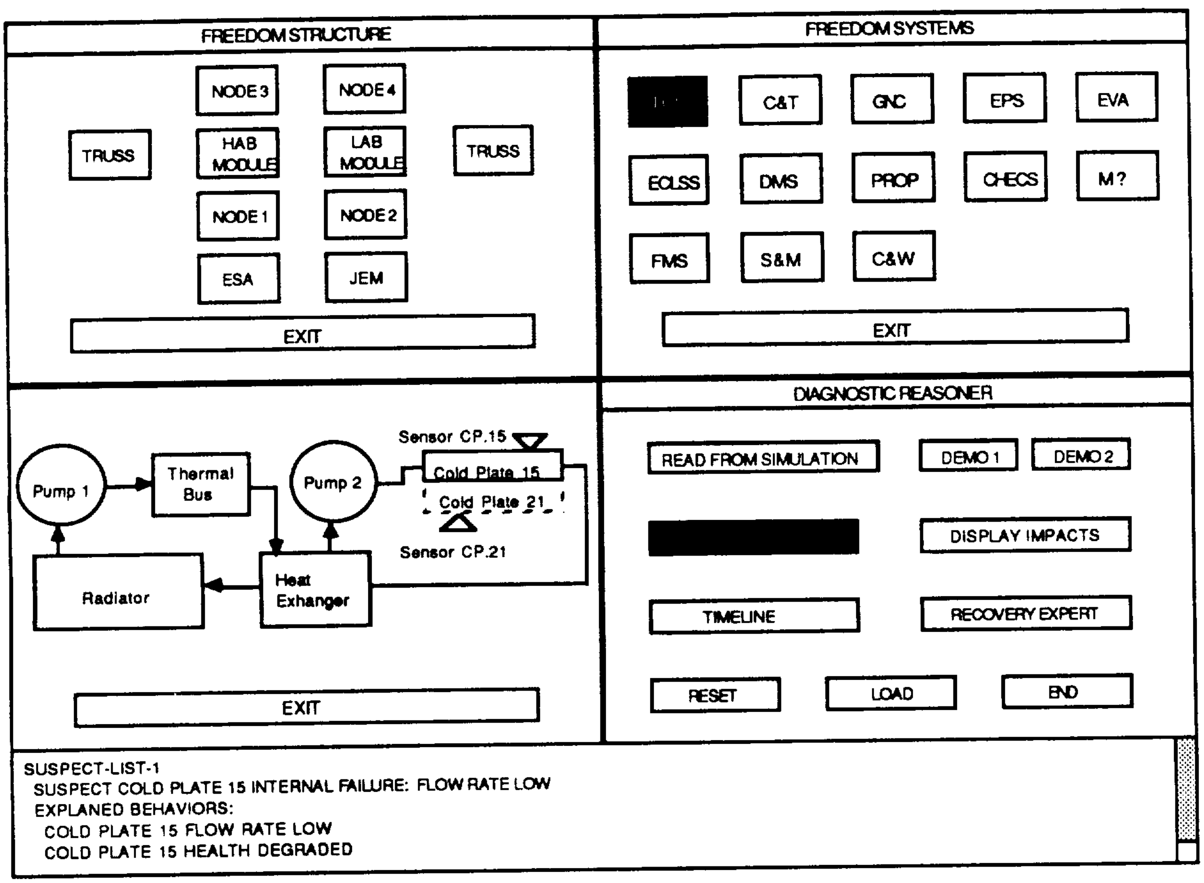


Figure 12-5. Display of Suspect List and System Component Graphic from Diagnostic Reasoner

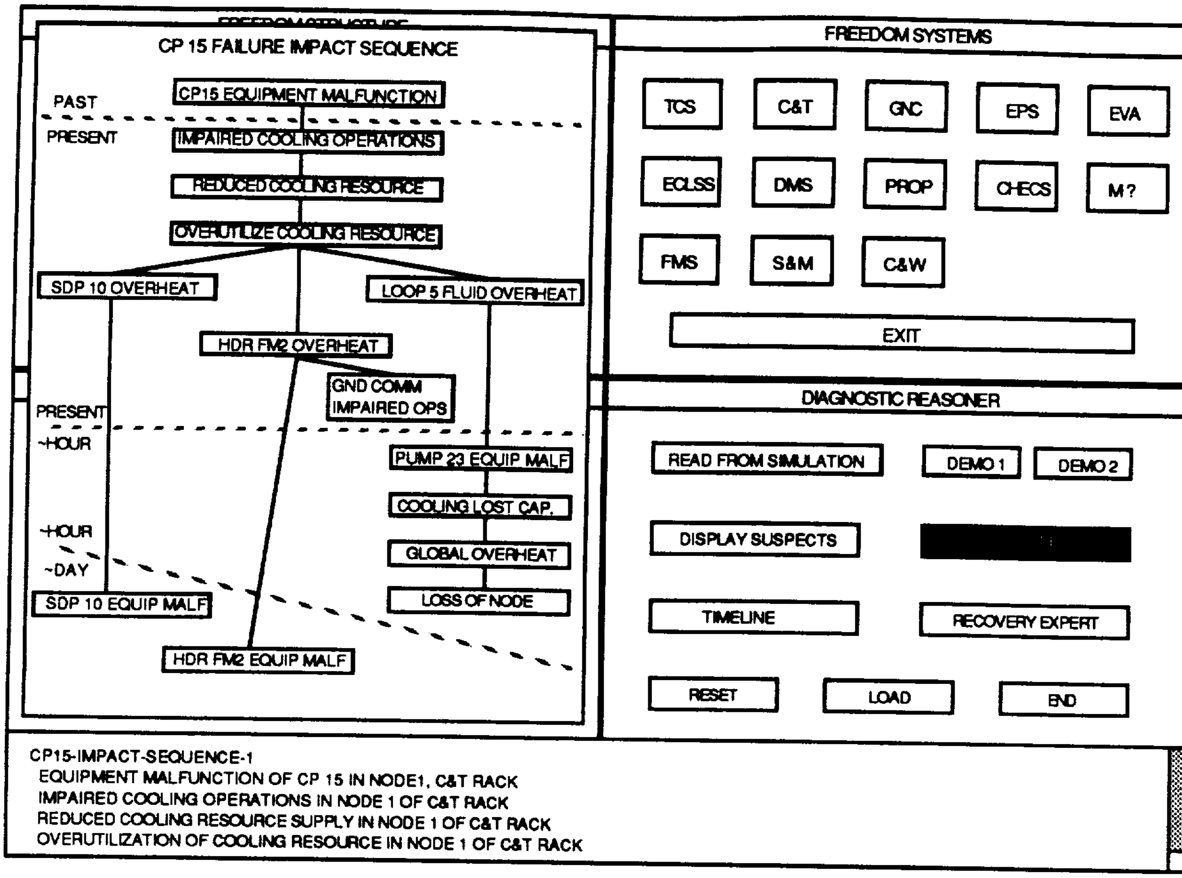


Figure 12-6. Fault Impact Sequence Display from Diagnostic Reasoner

The Timeline control button displays a window containing an event timeline summarizing major diagnostic events for each component of the system. A timeline is provided for each component. Anomalous events are indicated by color coding of the timeline (red - failed, yellow - cautionary, green - nominal).

The scenario postulated for the demonstration showed how two suspected problems could eventually be attributed to a single problem. Initially, a cold-plate failure is postulated. Later the frame muxer bit error rate is observed to be high and the operating temperature is too warm. Ultimately DR determines that the cold plate failure caused overheating of the frame muxer, resulting in its anomalous behavior.

Currently the system does not associate time with its conclusions. A planned enhancement is to add timetags to message list and to consider the sequence in which events occurred.

A significant limitation of the current display is the inability of TAE to dynamically alter a graphical form based on real-time data. Thus, some of the displays (e.g., timeline) serve as placeholders for future capability. Real-time graphical display is planned for later versions of the system, after the port to the X Window System has been completed.

Recovery Expert

The goal-activity network is the basis of the user interface (see figure 12-7). Diamonds represent goals and rectangles represent activities. Activities are equivalent to procedures used during mission operations. Activity names correspond to procedure names. A convention for naming goals remains to be defined. Normally, goals are gold. Blue is used to indicate a phantom goal, or a goal that has already been achieved. Network relationships are either conjunctive or disjunctive. There are two types of relationships: goal achiever (which relates a goal to an activity or a sub-goal) and pre-requisite. Conjunction and disjunction are distinguished using color. A statement of the activity is displayed within the rectangle. Selection of a goal causes a statement of the goal to be shown in the message panel at the bottom of the display. Selection of an activity provides the following menu options:

- Goal state resulting in procedure
- Entry and exit conditions
- Key binding (i.e., instantiation parameters)
- Pre-requisites

If a conflict exists, the reason for failure is accessible via this menu. If a homologue exists, it is indicated on this menu as well.

Both conflicts and homologues are identified on the network by distinctive icons. The conflict icon is a white exclamation point in a red box. Homologues are associated by a circle containing the same number. Since only one instance of the homologue is pursued to completion, the goal at the top of the pursued path is indicated by two concentric diamonds.

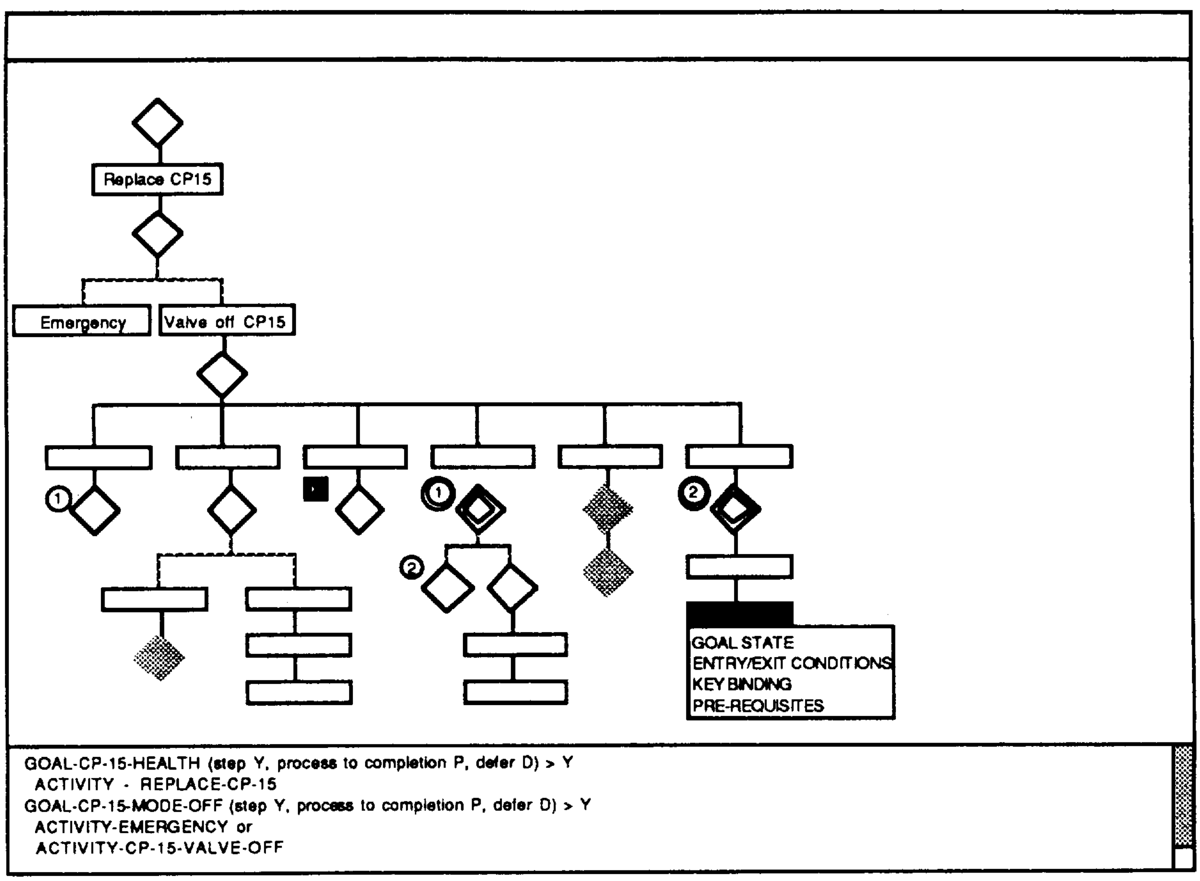


Figure 12-7. Goal-Activity Network from Recovery Expert

Control of Rx is initiated from a button panel. Control options include:

- Display Goal-Activity Network
  Display the goal-activity network (see figure 12-7)

- Display COA
  Display all possible COAs (see figure 12-8). This results in a popup window showing the name of each COA with an optimality score based on the key parameters (schedulability, timing, and undesirability).

- Generate Goal-Activity Network
  Initiates the creation of the goal-activity network. On-going results are displayed in the message panel at the bottom. Operator control inputs are also requested from the message panel, specifically step-wise generation of a path, pursue path until completion and deferral of a path.

- Generate COAs
  Evaluate the goal-activity network for all possible COAs

- Reset
  Reset the intelligent system

- Load
  Load the knowledge base

- Command Line
  Go to the ART command line

- Exit
  Exit the intelligent system

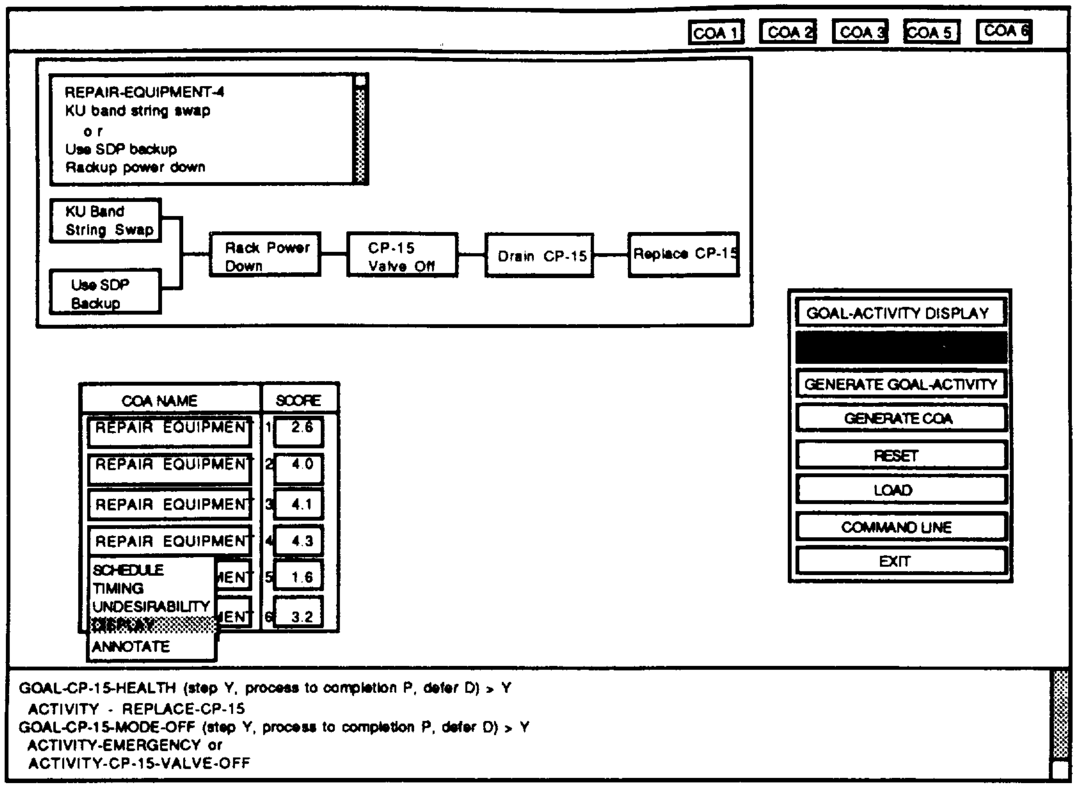


Figure 12-8. Course of Action Display and Rx Control Panel from Recovery Expert

When the COA has been displayed, selection of a button labeled with a COA name causes a menu to appear providing the following options:

- Schedulability
  Provides information about the schedulability portion of the COA score

- Timing
  Provides information about the timing portion of the COA score

- Undesirability
  Provides information about the undesirability portion of the COA score

- Display
  A window displaying the specific COA is popped up (see figure 12-8). Information includes a scrollable region containing a text description of the COA and a diagram illustrating the procedural flow of the COA.

- Annotate
  Provides the operator a mechanism for associated comments with the COA.

Procedures Interpreter

The workspace of the user interface is divided into three areas, (1) a header identifying the workstation and user, (2) a centrally-located dynamic region, and (3) a set of control options. The displays described in this section are all located in the dynamic region. The control options provide capability typically not used in real-time support (e.g., logging in and out of the system, accessing mail system, etc.).

The main control menu for the Procedures Interpreter provides the user with a variety of sources of procedures or Procedures Books, including:

- Pocket Checklist
- Operations Checklist
- Integrated Emergency
- Integrated Operations
- Maintenance
- Training
- Malfunctions
- Select from Individual Timeline
- Select from Crew Timeline
- Select from Space Station Timeline

Access to the Procedure Log and intelligent system control options (i.e., reset knowledge base, exit the intelligent system) are also provided from this menu.

Once a procedure has been selected and loaded into the system, it is displayed in the dynamic region as a sequence of text. If a procedure includes conditional activities (as expressed by "if-then-else"), all activities are displayed to orient the operator about the range of possible activities, not just the selected activities. A system prompt is located to the left of the currently active line. As the procedure executes, check marks are placed to the left of the completed line and a comment is appended to the right of the procedure (including timetag and status). If operator input is required, execution halts until it is provided. See figure 12-9 for an example of the PI display.

In addition to the annotated text of the procedure, the status of the procedure and relevant data plots can be viewed in the dynamic region. In the reboost example, altitude versus time is plotted. The status window describes the current activity (e.g., burn 2 in progress) and provides the values of relevant parameters (e.g., the time of ignition (TIG), the elapsed time in the burn, and the time until burn completion are all relevant to a burn procedure).

Alarms are enunciated near the bottom of the screen as a text message in bold print. They are also annotated in the body of the procedure with timetags.

Selectable commands are available at the bottom of most windows displayed in the dynamic region. These commands can include control of the intelligent system (e.g., restart, exit), control of the procedure execution (e.g., abort), control of the user interface (e.g., timescale), and control of input data (e.g., data).

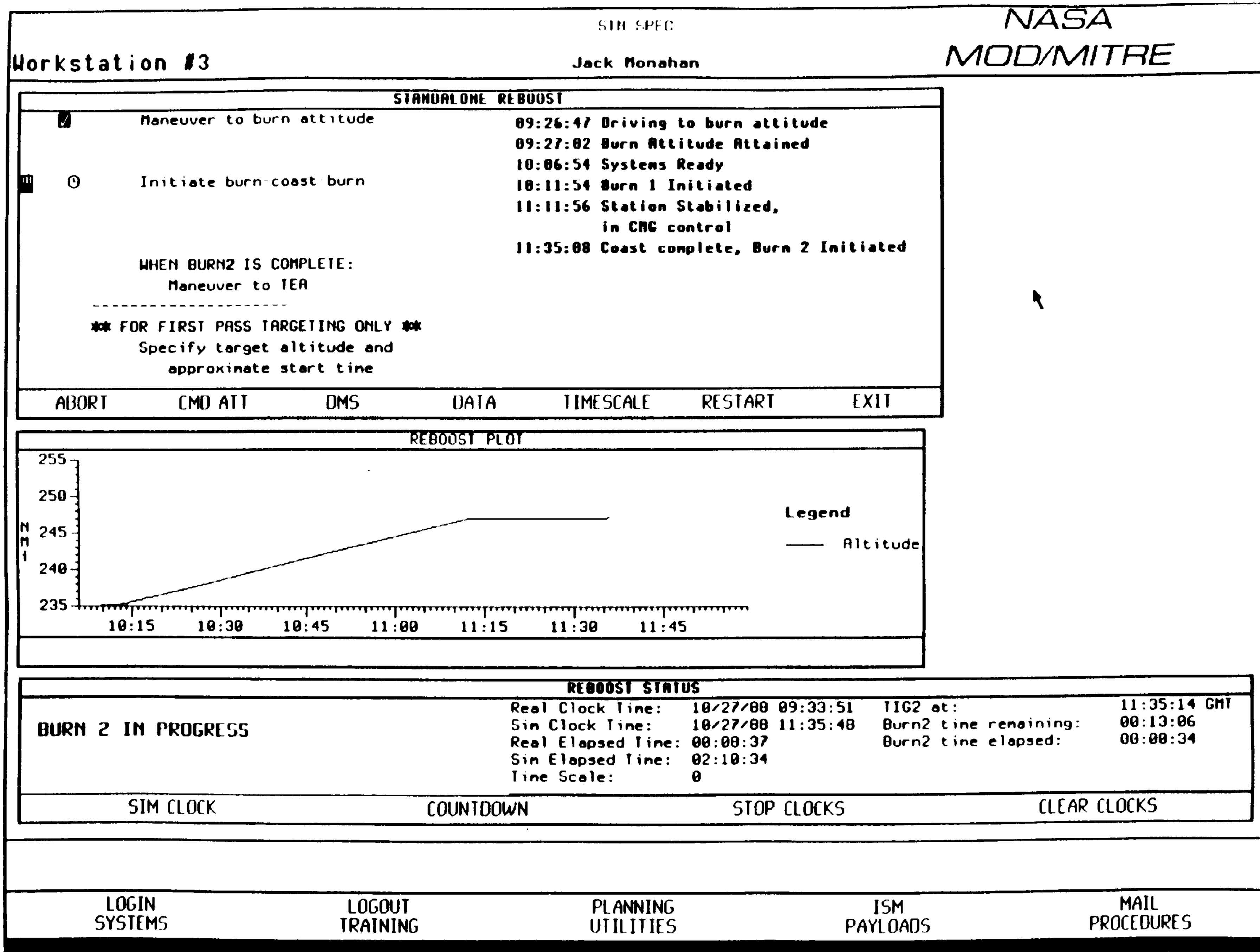


Figure 12-9. Procedures Interpreter Display

12.5 Design Process

The OMS prototypes were developed incrementally, using an iterative method of plan, assess, implement, and evaluate. They found the iterative approach useful in refining the prototype's capability, in upgrading the prototype based on user interaction with the prototypes, and in evolving requirements. This approach was consistent with the prototyping goals of understanding requirements and determining the nature and extent of user interaction with intelligent systems. The developers feel that early specification of complete requirements (as required in the waterfall development model) is difficult for highly interactive end-user applications, since requirements are often not well known or understood prior to prototyping (Baker et al., 1991).

12.6 Study Method

Information about the OMS Prototypes was obtained by interview of the project representatives and demonstrations of the prototype on January 19 and October 30 of 1990, and by review of the case data sources cited below. The displays used to illustrate DR and Rx are based on observations from a demonstration of these systems. These illustrations accurately characterize the user interfaces for both systems, but are not be exact duplicates of the actual displays. All project representatives are system developers.

Study Team

- Jane Malin (NASA Johnson Space Center)
- Debra Schreckenghost (The MITRE Corporation)

Project Representatives

- C. Jayne Guyse (The MITRE Corporation)
- Dave Hammen (The MITRE Corporation)
- Christine Kelly (formerly The MITRE Corporation)
- Christopher Marsh (The MITRE Corporation)

12.7 Case Data Sources

Baker, C. G., D. G. Hammen, C. M. Kelly, and C.A. Marsh (March, 1991), *The Operations Management application Failure Management Prototype*, WP-9100006, Houston, TX: The MITRE Corporation.

DSA (September, 1990), "TRIAGE Progress Report".

DSA (1990), "TRIAGE Test Scenario No. 2".

Hammen, David (May, 1990), "Recovery eXpert Prototype Design", briefing on May 19, 1990, Houston, TX: The MITRE Corporation.

Kelly, Christine (1990), hardcopies of displays from demonstration of Procedures Interpreter, Houston, TX: The MITRE Corporation.

Marsh, Chris, and Jayne Baker (May, 1990), "Diagnostic Reasoner Prototype Design", briefing on May 19, 1990, Houston, TX: The MITRE Corporation.

Marsh, Chris, et al. (August, 1990), "The Diagnostic Reasoner Prototype", briefing for the MITRE AI Cluster Group on August 23, 1990, Houston, TX: The MITRE Corporation.

Schreckenghost and Kelly (December, 1990), *Space Station Freedom Program Advanced Automation: Volume II Evolution within the Test Beds*, MTR-89W00271-02, Houston, TX: The MITRE Corporation.

Part II

Case Study Performed by Study Team from Ohio State University

Section 1
Introduction

Part II contains the reports from the case study performed by the study team located at the Ohio State University (OSU). The following NASA fault management systems are described in part II:

- Space Station Module/Power Management and Distribution (SSM/PMAD) at Marshall Space Flight Center (MSFC)
- Space Station Human Interface to the Thermal Expert System (HITEX) at Ames Research Center (ARC) and Johnson Space Center (JSC)
- Spacecraft Health Automated Reasoning Prototype (SHARP) at Jet Propulsion Lab (JPL)
- Space Shuttle Knowledge-based Autonomous Test Engineer (KATE) at Kennedy Space Center (KSC)
- Space Shuttle Intelligent Launch Decision Support System (ILDSS) at KSC

These case studies were constructed based on demonstrations and interviews with systems developers and/or available documentation on the projects. Due to the variability in the availability of these resources, there is considerable variation in the completeness of these case studies.

These reports represent a single time-slice of information for systems that may be undergoing revisions. Therefore, these case study reports may not reflect the present state of these systems.

Section 2
Space Station Module/Power Management and Distribution System (SSM/PMAD)

This chapter is a result of a unique opportunity to study the development of a user interface over a four month period of time. In July of 1990 the study team received a demonstration of the system and its capabilities. At the request of the PMAD designers, Cognitive Systems Engineering Laboratory (CSEL) provided input to improve several aspects of the interface capabilities, as that part of the system was undergoing a general revision and upgrade. A second site visit was made in November of 1990 to examine the upgraded interface. This allows the report on this system to contrast before and after on several aspects of the human-intelligent system interface. The structure of this document will attempt to convey the changes that were made. This will be accomplished by identifying 'before' and 'after' descriptions. Paragraphs pertinent to one of the demonstrations will be preceded by the identifier (7/90) or (11/90). Items that did not change (such as the monitored process or parts of the interface) will not be preceded by this identifier.

2.1 System Description

SSM/PMAD provides autonomous system management and contingency rescheduling for the module level power distribution of the Space Station's power resources both during normal and faulted operation. The SSM/PMAD intelligent system is comprised generally of the scheduling, load priority management and diagnostic systems. In addition, there is substantial algorithmic software throughout the system.

The scheduling system consists of MAESTRO, the master scheduler, and Front End Load Enable Scheduler (FELES), the front end to the Scheduler. The Load Priority List Management System (LPLMS) comprises the load priority management system. The Fault Recovery and Management Expert System (FRAMES) is the diagnostic system. See figure 2-1 for a simplified version of the data flow between these systems. The actual power hardware consists of Remote Power Controllers (RPCs) and Remote Bus Isolators (RBIs) which are controlled by Generic Controllers (GCs). These are described in detail in the next section.

Monitored Process

The power distribution hardware consists of several load centers and two power distributors, each having several remote power controllers (RPCs) which deliver power through two busses to the various loads. A RPC is comprised of a power stage and a generic controller (GC) card. The power stage provides current limiting during a turn-on in-rush or during a load fault. The GC card accepts analog current, voltage, and temperature signals from the power stage and supplies I^2t, undervoltage, and overtemperature trip capability for the RPC. An RPC acts as a power system circuit breaker by safing the system immediately following a fault.

Lowest Level Processors (LLPs) turn the RPCs on or off according to a downloaded schedule. They also monitor the sensors and RPCs and notify the diagnostic system (FRAMES) if a trip occurs, and if an RPC is using more power than it is scheduled to use. The kind of trip is reported to FRAMES as well.

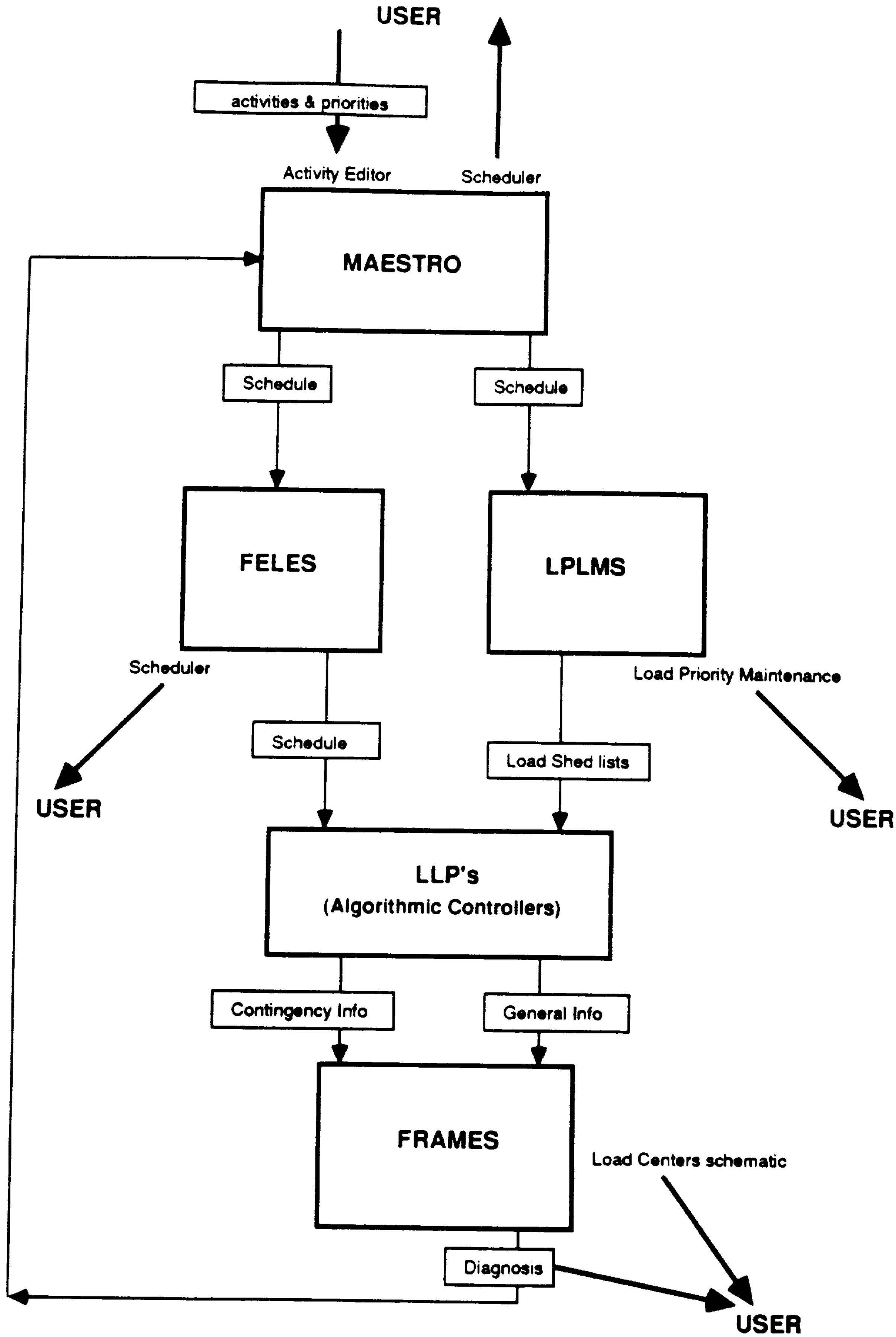


Figure 2-1. System Data Flow

If a faulted RPC is marked as redundant, the LLP will turn on the redundant RPC. Switching a load to redundant power may force other loads on the redundant bus to be shed due to limited power availability and their relative lower priority. FRAMES is informed of such changes as well. Each LLP stores a priority list of its loads which is handed down by the LPLMS. FRAMES informs the scheduling components of such changes so that they will be reflected in the schedule.

Man-Machine System

When building the primary power enable schedule, the user defines activities by subtasks, specifying maximum power limits, initial properties, and redundant power requirements. The user enters this information in the Activity Editor in MAESTRO (see figure 2-1 -- the darker arrows indicate user interfaces, and the name between the arrow and box indicates the name of the interface; the smaller boxes indicate the information that is passed among modules).

Currently when a fault occurs, interrupting power to a load, the load is repowered only if it has redundant power capability. If there is no redundant power for a load whose switch has failed, that load is shed. These changes are sent to the scheduling components.

Users should be able to dynamically adjust schedules and have the adjustments reflected at the scheduler. This is planned as a future enhancement.

Development and Testing Environment

The scheduling system (MAESTRO, FELES) and LPLMS reside on a Symbolics™ 3620D. The diagnostic system (FRAMES) resides on a Solbourne (SUN® clone). The test environment includes three load centers, two power distributors, and a communications interface between them and the higher level controllers, e.g. FRAMES.

There are plans to add an independent fault injection capability in order to test SSM/PMAD. Currently, however, shorts are manually applied to cause switches to trip.

2.2 Intelligent System and Functions

FRAMES

FRAMES diagnoses single and multiple hard faults. Examples of hard faults include: open or short circuits, and different kinds of trips. FRAMES also diagnoses masked faults. (7/90) Multiple simultaneous faults were not considered likely, and diagnosis of such faults was not implemented. (11/90) Multiple fault diagnosis capability has been recognized as important, and implemented through a new knowledge control mechanism which has lead to a new fault isolation methodology.

Soft faults are not diagnosed because they require increased precision in the data produced by A/D data conversion. Also, processes handling the non-synchronicity of the distributed control system are presently candidates for soft fault research.

FRAMES matches observed symptoms with a symptom-set fault table in its knowledge base. FRAMES monitors several parameters such as current, voltage, temperature and power of

™ Symbolics is a trademark of Symbolics, Inc.

® SUN is a registered trademark of Sun Microsystems, Inc.

individual sensors as well as current, state, tripped status and power availability of switches. It performs a type of diagnostic intervention, switch manipulation, in order to test hypotheses about faults. However, as the contractor's documentation notes, this tool is limited in that it sometimes cannot isolate the cause to an individual component.

FRAMES uses rule-based reasoning. (7/90) Model-based reasoning is not considered necessary but may be implemented in future releases to account for soft faults. (11/90) The possibilities of including a KATE-based (Knowledge-based Autonomous Test Engineer -- KSC) expert system (model-based reasoning) into the SSM/PMAD test bed are to be explored. A model-based reasoning system will potentially be developed to perform fault analysis in parallel with the rule-based FRAMES. Then, KNOMAD may potentially be able to integrate the best of both methods. Note: this is only being discussed at this point.

MAESTRO

The only type of rescheduling that occurs now in MAESTRO is load shedding and transfer to a redundant power source. A fault will cause a load to be shed, unless a redundant power source has been supplied for it.

While many interesting issues are raised by this scheduling system, we chose not to assess the interaction style and capabilities used in MAESTRO due to limited access to the system.

LPLMS

The LPLMS passes a priority list of loads to the LLPs every 15 minutes. The criteria used to order the priority list takes into account what each RPC is being used for, how much power it is drawing, and how important it is to provide power to the task. Also, since several different tasks may use that RPC during the 15-minute interval, the influence of all the tasks is taken into account in the assessment of whether to shed a particular RPC.

2.3 Human-Intelligent System Interaction Functions

Assessment

There are three general categories of information that provide the user with an assessment of the current global situation. These are: power situation information, parameter values, and intelligent system state information.

By power situation information, we mean information about what the current schedule is, what loads are on what switches, what switches are failed, etc. By parameter values, we mean the values on sensors, and switches that are monitored.

(7/90) The different states that the SSM/PMAD can be in are: autonomous and manual mode, and within autonomous mode, the system may be in the following states: monitoring, diagnosing, and making schedule changes.

(11/90) Plans have been made for a mixed mode of operation. However, it was not implemented at this time so could not be studied.

1) Power Situation Information

(7/90) The information on what switches are failed and whether a load has been transferred to a redundant source is present all the time on the FRAMES schematic display. In order to find out what activity is represented by what load the user must go to the FELES screen and mouse on a particular RPC. If the user would like to know what RPC a particular activity is on, he must call up the LPLMS screen.

(11/90) Failed switches are represented in the power system screen (what was previously called the FRAMES schematic display). There has not been any solution to the problem of determining what activity is represented by which load. The possibility of a function- based representation is presented later.

The user may also find out what RPCs are being used by what activities by selecting the scheduled activities from the MAESTRO scheduler screen.

If FRAMES discovers a fault, and a schedule update is made, the user is not alerted to the fact that a revised schedule has been issued. The new schedule will replace the previous one when the user calls up the MAESTRO screen, but information highlighting the change is not provided.

2) Parameter Values

(7/90) Current, voltage, temperature and power of individual sensors as well as current, state, tripped status and power availability of switches (RPCs) is available to the user via the FRAMES schematic display.

(11/90) Current, state, and status of switches is available to the user via the power system screen. There was discussion of providing additional information when the user selects an expanded view, but at the present time the schematic was simply enlarged to occupy the entire CRT.

3) Autonomous vs. Manual

(7/90) The user is informed of autonomous vs. manual mode by the word 'autonomous' or 'manual' on the bottom of the screen.

(11/90) Present mode (manual, autonomous, or mixed) is displayed in the title bar at the top of the screen. The status indication is designed to provide better feedback than the previous version by being more salient (by being in the upper part of the workspace and in larger character size). However, it was suggested that additional techniques be used to differentiate mode (e.g., additional emphasis on this indicator or 'greying' of options that are not available in a particular mode; the latter would be consistent with their technique of indicating unavailable options in the icon selection menu and consistent with the results of Johns (1990)).

4) Diagnosis mode

(7/90) The user can tell when a diagnosis is in progress because a message appears in the FRAMES message box. The message informs the user that an LLP has reported a fault or that FRAMES is performing a test on an RPC. The user can tell when a diagnosis has occurred because the message 'Finished diagnosing' appears in the message box. In order to see the actual diagnosis, the user must select the item 'Diagnosis Window' from the menu, 'SUMMARY.' However, as a future enhancement, this window may be set up to appear automatically upon completion of a diagnosis. The diagnosis window presents the results of switch manipulation and specifies the most likely and least likely faults (see Diagnosis Window in figure 2-2).

Frames Status Messages
opening switches: (D18)
Collecting data from LLPs (snapshot)
flipping switch: D18
Collecting data from LLPs (snapshot)
LLP LLP-D reports D18 tripped on FAST-TRIP.
opening switches: (C18)
Collecting data from LLPs (snapshot)
flipping switch: C18
Collecting data from LLPs (snapshot)
LLP LLP-C reports C18 tripped on FAST-TRIP.
LLP H reports a fault.
Collecting data from LLPs (snapshot)
LLP H reports a fault.
Collecting data from LLPs (snapshot)
LLP A reports a fault.

Diagnosis Window
D18 tripped on FAST-TRIP.
During testing D18 retripped on FAST-TRIP.
POSSIBLE CAUSES:
Most Likely:
Low impedance short in cable below switch, switch output of switch, or the switch input of one of the lower switches.
Less Likely:
Current sensor in switch reading high.
C18 tripped on FAST-TRIP.
During testing C18 retripped on FAST-TRIP.
POSSIBLE CAUSES:
Most Likely:
Low impedance short in cable below switch, switch output of switch, or the switch input of one of the lower switches.

Figure 2-2. FRAMES Selectable Windows

(11/90) The diagnosis window is present at all times (except when the power system screen is expanded to a full screen view). The diagnosis window now includes the status window messages. These are separated from diagnosis messages by space and horizontal lines in an attempt to aid the user in understanding the messages (see figure 2-3).

(7/90) If a diagnosis has been performed, and the user leaves and later returns, it will not be apparent how many diagnoses and faults have occurred within that time period. The user must remember the previous state and/or call up the status and diagnosis windows to find out in detail about the potential new faults. Currently there is no mapping between the time of occurrence and the diagnostic events.

(11/90) Previous diagnoses can be viewed through the implementation of scrollable windows. The scroll bar on the window is only active when text has scrolled off the screen (which aids in indicating when hidden diagnoses exist).

5) Monitoring mode

Monitoring is considered the default mode. Therefore if there is no diagnostic message in the message box, FRAMES is in monitoring mode.

6) Schedule Changes

Users are not directly informed of rescheduling events. The user must call up the MAESTRO Scheduler screen in order to examine the current loads.

Collaboration

There is no explanation of the diagnosis and the user interface does not provide access to a fired rule trace. The system also does not inform the user of components it does not consider in its reasoning.

(7/90) In order to see the tests that FRAMES has performed, the user must select the 'FRAMES Status Messages Window' from the SUMMARY menu.

(11/90) Tests performed are presented in the diagnosis window, as mentioned above.

The user can request some specific data, such as current, state, status and power availability of individual switches. However, the user may not choose tests to perform, or control the flow of diagnosis. Diagnosis is autonomous rather than collaborative/cooperative.

(7/90) The demonstration did not show how or if the user can play an active role in the rescheduling process. User rescheduling which was intended to be part of the design, may be implemented in future releases.

Currently no editing capability for entering activities exists. The user inputs schedule information, activities and the priorities associated with each activity via the Scheduler interface. If a change is desired, a new schedule must be entered. An editing function may be implemented in future releases.

```
Mode is autonomous.
RPC D18 tripped on FAST-TRIP.
opening switches: (D18)
flipping switch: D18
RPC D18 tripped on FAST-TRIP.
---------------------------------------------------------------


15
D18 tripped on FAST-TRIP.
During testing D18 retripped on FAST-TRIP.
POSSIBLE CAUSES:
Most Likely:
  Low impedance short in cable below switch, switch output o
f switch, or the
switch input of one of the lower switches.
Less Likely:
  Current sensor in switch reading high.
Finished making diagnosis.
```

Figure 2-3. Modified Diagnosis Window

Intervention and Take Over

The system operates in strictly autonomous role in which information is available only as display, or in a strictly manual mode. Manual mode allows the user to turn all the switches on or off. The user also has access to all the breadboard sensors. When in manual mode, the system does not the aid user in discovering the implications of his actions; the AI functions 'cease operation within the system'. In this mode user changes are simply reflected in the affected components. However, there are plans to add intermediate levels of autonomous function in future releases.

2.4 Supporting User Interface Capabilities

Workspace

(7/90) Figure 2-4 shows how the workspace is organized. The Scheduling system (MAESTRO and FELES) as well as LPLMS screens are on one black and white monitor and the FRAMES screen is on a separate color monitor.

(11/90) Figure 2-5 presents the revised workspace. The MAESTRO interface has not changed, but considerable modifications have been made to the components presented on the color monitor (previously called FRAMES interface). The present configuration consists of a seven-screen interface using tiled windows. The different screens are:

- Power system
- FELES
- FRAMES
- LPLMS
- Communications
- Power Utilization
- System Flow

The FRAMES, Power Utilization, and System Flow screens have not yet been implemented. Of the remaining four, the power system screen received the most attention and will be discussed in the most detail.

Information and Presentation

(7/90):

The FRAMES user interface consists of a single background window, or screen. This is the screen showing a physical mimic (schematic) of the power distribution system, i.e. the PDCUs and load centers. However, other windows displaying diagnostic messages and parameter values pop up in this window via menu selection. When these pop up they may cover other information on the screen, such as the flow diagram of the power distribution system, but they may be moved by the user to a 'blank' location on the screen.

The information in the FRAMES Status Messages Window and the FRAMES Status Window is presented as a running stream of text (see figure 2-2).

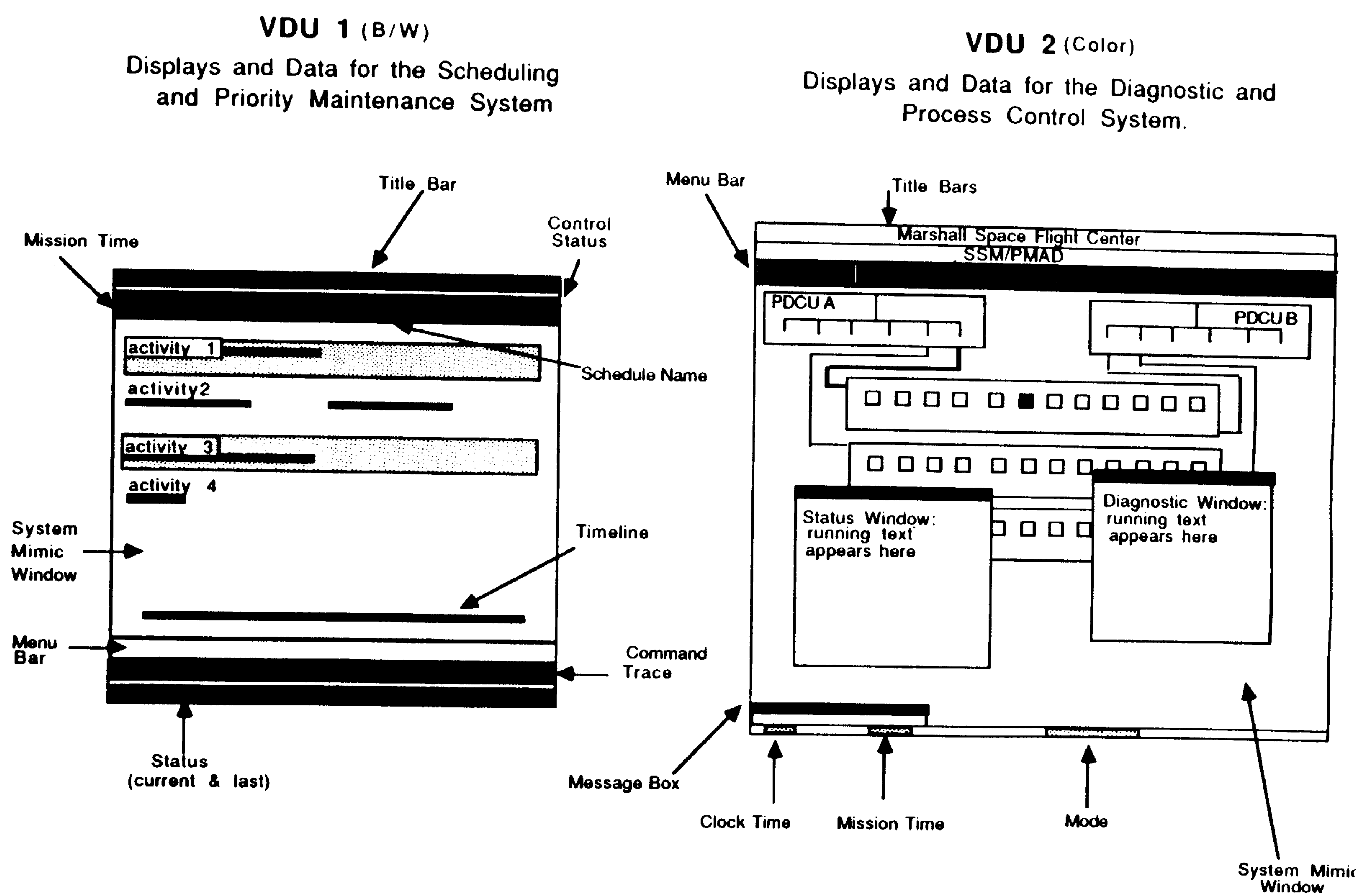


Figure 2-4. Workspace Level View of the User Interface to SSM/PMAD

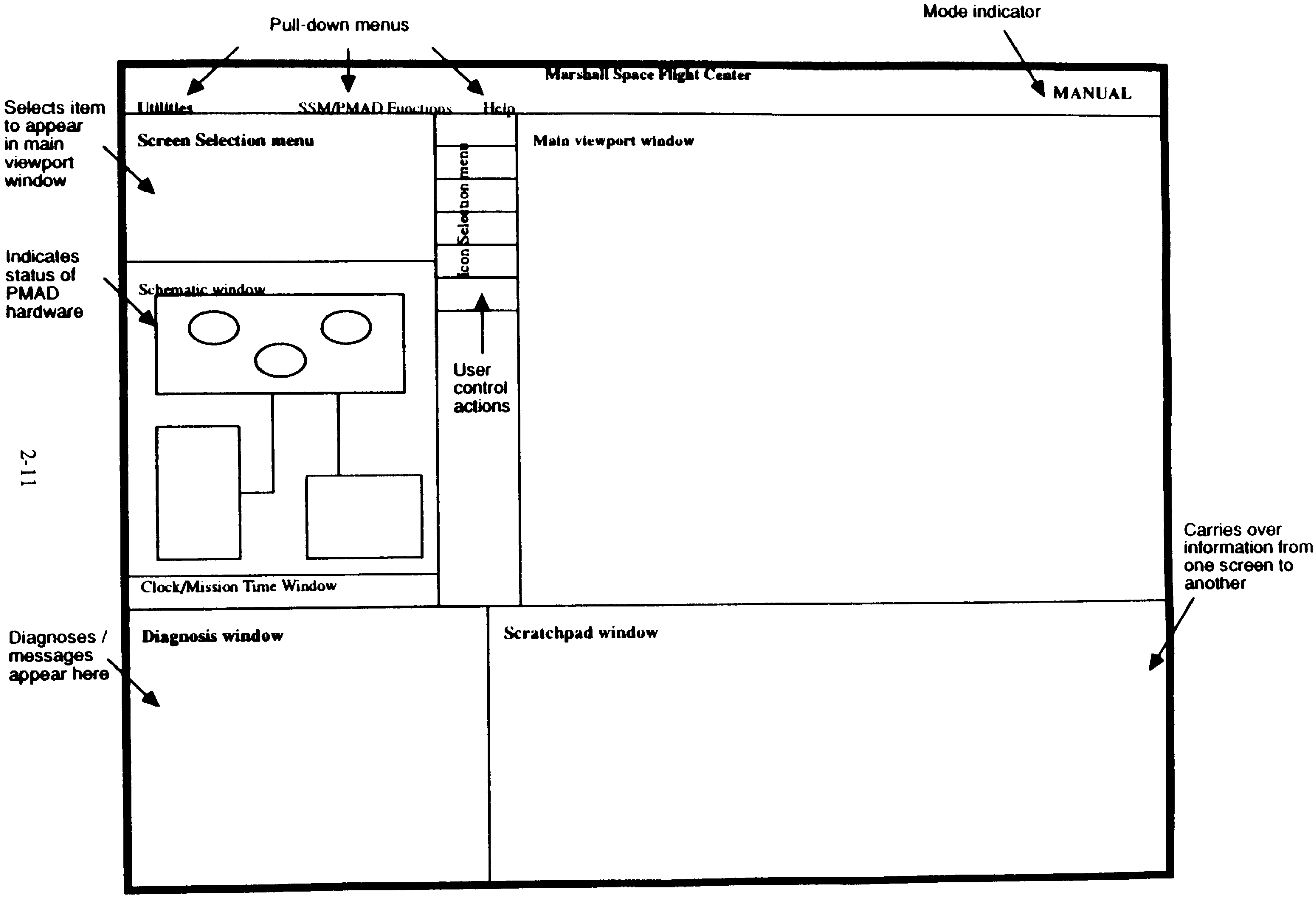


Figure 2-5. Revised Workspace Level View of the User Interface to SSM/PMAD (right Visual Display Unit)

Status of individual switches in FRAMES is displayed via color coding. Normal, under test, and fault states are represented with a different color. Out of service switches are indicated by black. Switches reporting fault conditions which are under test and are not out of service are indicated in red. There is also a representation of the load centers and switches in FELES, however, in this case a different coding is used to represent the status of the RPCs; black is used to indicated normal, and faulted switches are indicated by a symbol.

The parameter values of switches and sensors are displayed by clicking on the representations of these items on the FRAMES screen. For example, if the user selects the icon of a sensor on load center #1, a small green window pops up that displays the voltage, temperature, current and power. These values are displayed digitally and only the current values are shown. The window remains on the screen until the user clicks on the 'close' box.

MAESTRO displays each activity of the schedule as a rectangular block mapped on a timeline. The 'mission time' is displayed in the upper left hand corner of the screen. The time is updated every minute (see figure 2-4).

(11/90):

Discussion centered around the power system screen, which provides a dynamic schematic of the load centers and RPCs. This screen displays the status of individual switches by dynamic icons (coded such that open and closed positions are consistent with an actual switch, making switch position easily detected) and the status of RPCs by color coding (green to indicate OK, red to indicate a fault detected, and black to indicate taken off line). Active path of flow is indicated by flow path coding (when current is present, path fills in yellow). Additionally, location of sensors is indicated by small circles.

There was some discussion as to the problems in expanding the present system to accommodate more than five load centers. While the test version was designed so that five load centers would fit in the window, the real system (with more than five load centers) will require additional capabilities. Possibilities include pan/zoom and center/surround. Implications of these approaches would need to be carefully considered.

One issue that is not addressed in this representation is how to select what switches to load. Different components are linked so that placing additional load on certain RPCs is more likely to overload one subsystem than others. There needs to be some representation of this problem in order to aid the operator in selection of RPCs. One approach (as mentioned earlier) would be to switch to a functional, rather than physical representation.

Support for Interaction

1) Modes

(7/90) In FRAMES automatic or manual mode is indicated by the word 'automatic' or 'manual' in a relatively small font at the very bottom of the screen. It was often the case that we and the NASA personnel showing the demo did not know what the system was doing, because insufficient data is provided, especially data about events and change.

(11/90) See 'Autonomous vs. Manual' in section 2.3. Within manual mode, components can be manually selected for testing (by pointing and clicking with the mouse). An added feature is the use of graphics to place a check mark on the component when selected. This allows the user to easily keep track of which components have been selected.

2.5 Design Process

(7/90):

The project was contracted to Martin Marietta. We met only with the NASA project personnel. It appears that there was minimal up-front definition of the user interface requirements; instead the development of the user interface seems to have been ad hoc.

NASA project personnel did not think potential users (e.g. process engineers) were consulted in order to test aspects of the user interface or that any rapid prototyping of the user interface was done. Furthermore, it appears there was no human-computer interaction specialist included in the contractor's part of the design team to consider the user interface.

During our briefings the view was expressed that the development of the user interface should be left until the end because the system could be expected to change during the several years of development, thus avoiding 're-doing' the work.

NASA project personnel expressed the feeling that the user interface was weak but felt unable to identify specific features that should be changed or added to make the interface stronger.

(11/90):

We met with NASA as well as the Martin Marietta project personnel. Significant attention to the user interface had been devoted since (7/90), incorporating many of features based on a discussion of issues on (7/90).

2.6 Summary of Issues

- Workspace Navigation

  The number and flexibility of windows in SSM/PMAD has the potential to create navigation difficulties and dissociation of related process views. In the 7/90 interface, a background schematic display was overlaid with windows including status messages, diagnostic messages, and parameter values. In the 11/90 version, parameter values were integrated into the schematic display and output from the diagnostic system (status and diagnostic messages) were presented in a non-overlapping window. The power system screen (physical schematic) was one of seven screens which could be presented in serial. Since several of these screens were not finished, a full assessment of the navigation issues could not be performed. More information can be found in section 5.2 (Workspace: Proliferation of Windows) in Volume 1 (Malin et al., 1991).

- Physical Topology Schematic Display

  SSM/PMAD uses a schematic display of the load centers and RPCs. In 7/90 this was predominantly a static representation, with faulted components being indicated by color coding. As indicated in section 5.4 (Physical Topology Schematic) in Volume 1 (Malin et al., 1991), this approach may not adequately highlight events and anomalies in the process. In 11/90 this schematic display was made more dynamic through flow path coding and dynamic switch icons.

- Message Lists

  SSM/PMAD was very similar to several of the other cases investigated in that intelligent system output was in the form of chronologically ordered message lists. In an attempt to assist operators in interpreting the messages, status and diagnostic messages were separated by vertical space and horizontal lines (in the 11/90 interface). However, this approach does not appear to provide sufficient information about temporal characteristics, type of events, and type of intelligent system activity. Volume 1 (Malin et al., 1991) contains substantial information on this topic.

2.7 Study Method

Study Team

- David Woods (Ohio State University) (7/90 & 11/90)
- Leila Johannesen (Ohio State University) (7/90)
- Scott Potter (Ohio State University) (11/90)
- Michael Shafto (NASA Ames) (7/90)

Project Representatives

- Louis Lollar (NASA MSFC) (11/90)
- Dave Weeks (NASA MSFC) (7/90)
- Bryan Walls (NASA MSFC)(7/90)
- Barry Ashworth (Martin Marietta) (11/90)
- Michael Elges (Martin Marietta) (11/90)
- Laura Jakstas (Martin Marietta) (11/90)
- Chris Myers (Martin Marietta) (11/90)
- Joel Riedesel (Martin Marietta) (11/90)

Components Studied

(7/90) While we looked at MAESTRO, FELES and LPLMS, we concentrated mainly on the FRAMES user interface.

(11/90) As mentioned previously, the MAESTRO, FELES, and LPLMS interfaces had not changed. Additionally, the new FRAMES, Power Utilization, and System Flow interfaces had not been implemented. While the Communications interface was implemented, no attention to HCI issues had taken place. Therefore, the power system interface and overall interaction support were the only areas addressed.

Process

Demonstrations of the overall system were done by using the test facility to induce power anomalies and observing FRAMES' response and the implications for the scheduler. The faults induced in the demos were hard faults, e.g. a tripped switch.

Individual screens served as departure points for examining details of the system and discussing 'what if' scenarios. Also, to a lesser extent, we tried entering data and moving from screen to screen.

2.8 Case Data Sources

Ashworth, B., J. Riedesel, C. Myers, L. Jakstas, and D. Smith[1] (July, 1990), *Space Station Automation of Common Module Power Management and Distribution -- Interim Final Report*, Martin Marietta Aerospace Denver Astronautics Group Report MCR-89-516.

Malin, J. T., D. L.Schreckenghost, D. D. Woods, S. S. Potter, L. Johannesen, M. Holloway, and K. D. Forbus (September, 1991), *Making Intelligent Systems Team Players: Case Studies and Design Issues, Volume 1. Human-Computer Interaction Design*, NASA Technical Memo, Houston, TX: NASA - Johnson Space Center.

Miller, W., E. Jones, B. Ashworth, J. Riedesel, C. Myers, K. Freeman, D. Steele, R. Palmer, R. Walsh, J. Gohring, J. Pottruff, J. Tietz, and D. Britt (November 1989), *Space Station Automation of Common Module Power Management and Distribution*, NASA Contractor Report 4260, Martin Marietta.

Riedesel, J. (February, 1990), *An Efficient Distributed Architecture for Cooperating Expert Systems*, Research Report.

Walls, B. (August, 1989), "Exercise of the SSM/PMAD Breadboard", *Proceedings of the 24th Intersociety Energy Conversion Engineering Conference*, Washington, D.C., pp.189-194.

[1] This report contains copies of recently published papers relevant to the SSM/PMAD testbed.

Section 3
Human Interface to the Thermal Expert System (HITEX)

3.1 System Description

HITEX has two roles: It is the user interface to the Thermal Expert System (TEXSYS) , which controls and diagnoses faults in the Boeing Aerospace Thermal Bus System. Also, it directly displays information about the thermal bus system via the Thermal Data Acquisition System (TDAS). (TDAS is the lowest level of information about the thermal bus system available to the engineers.) HITEX itself does not have any direct control over the thermal bus system, but simply functions as a channel for control actions through the expert system. If the expert system should become inoperative, HITEX can still display information from the TDAS for monitoring purposes.

Monitored Process

The thermal bus is a system of evaporators, pumps and condensers that transport heat from one location to another. The thermal bus used in TEXSYS is a two-phase system. In such systems liquid must be kept at a certain temperature and pressure. Heat is applied to an evaporator surface which vaporizes some of the liquid coolant. This vapor is transported to condensers where heat is given off and vapor is returned to a liquid state. Thermal engineers are concerned with monitoring the temperature, pressure and flow in the vapor and liquid lines. Some of the major classes of faults are: evaporator dryout, valve failures, leaks and faults in the pumping mechanism. Typical procedures that the thermal engineer might perform are: start-up, shut-down, and set-point temperature change. The time constant for most faults is on the order of tens of seconds, or minutes.

Man-Machine System

The primary users are thermal engineers. Although astronauts would monitor the system in the Space Station, the HITEX interface was geared toward providing a greater level of detail than they would need in order to evaluate the hardware for the Space Station.

The expert system controls the thermal bus system. Operator initiated commands are sent to TEXSYS which represents the change of state and sends the commands to TDAS. TEXSYS sends data to HITEX concerning fault diagnosis, procedure execution, or in response to operator initiated actions.

Development and Testing Environment

HITEX uses the following software tools:

- Genera 7.2® (a superset of Common LISP) by Symbolics, Inc.
- KEE™ 3.1 by IntelliCorp
- Color KEEpictures™ by IntelliCorp
- Tools of the Schematic ToolKit (STK)

® Genera 7 is a registered trademark of Symbolics Inc.
™ KEE is a trademark of IntelliCorp.
™ KEEpictures is a trademark of IntelliCorp.

HITEX and TEXSYS run on separate Symbolics™ 3650 computers. TEXSYS was based upon Intellicorp's KEE, and uses its truth maintenance capability. TEXSYS also uses two toolkits built by ARC on top of KEE: the Model ToolKit (MTK), which allows for rapid construction of qualitative models of physical systems, and the Executive ToolKit (XTK), which allows tasks initiated by the expert system to have a predefined range of responses to changing bus conditions.

Besides TEXSYS and HITEX, the overall system architecture consists of the thermal bus, the data acquisition and control system, and the TEXSYS data acquisition system (TDAS). TDAS is implemented in C on a stand-alone MicroVAX™ .

3.2 Intelligent System and Functions

TEXSYS is a model-based expert system based on KEE's software. It uses a fault diagnostic approach based on that of deKleer and Williams (1986; see Remington and Shafto, 1990) and uses KEE's truth maintenance capability. When a deviant sensor reading is detected all faults associated with it are kept active. Diagnosis progresses with the truth maintenance system pruning all those possibilities inconsistent with the evidence.

When a deviant value is detected, TEXSYS establishes a goal, which is usually to find the fault. Subgoals and then tasks are spawned. The functions of tasks include performing sensor comparisons, generating tests and informing the user. This goal-based approach is also used to execute procedures. The rules used for executing procedures are context sensitive.

In summary TEXSYS performs:

- Automatic control of nominal and off-nominal procedures
- Data monitoring for fault detection, isolation and recovery with respect to current operational mode
- Data monitoring for fault prediction via trend detection and analysis.

3.3 Human-Intelligent System Interaction Functions

Assessment

The different kinds of data available are: sensor readings, valve states, component status, task status, faults diagnosed, and warning messages. Details on how the data is presented will be given in section 3.4 on Supporting User Interface Capabilities, in the context of the available displays.

Status of the thermal bus system components on the graphics screen is indicated four ways:

- The component color changes to red when the status is not nominal
- If the component is a plotted sensor, its line color will change if the sensor is off-nominal

™ Symbolics is a trademark of Symbolics, Inc.
™ MicroVAX is a trademark of Digital Equipment Corporation.

- Selecting status from the component menu produces a message about the anomaly
- A warning message is displayed

(If a sensor has been "undisplayed" by the user, an anomaly causes it to reappear.)

Users are given information about the expert system's diagnosis and reasoning after the diagnosis is completed. A graphical justification tree shows the progress of fault diagnosis. An operator may query the system to find the current hypothesis and to get a list of the tasks applied at any point in time and the rules used to invoke tasks and subgoals.

Collaboration

The operator may initiate the following predefined procedures:

- Setup NCG Venting Parameters
- Initiate Setpoint Change
- Cut off all Heatloads
- Open/Close Valves
- Toggle a Heater
- Turn on/off RFMD
- Off Nominal Shutdown
- Check RFMD Voltage and Frequency
- Activate/Turn off Sensor
- Nominal Startup/Shutdown
- Vent NCGS from RFMD Once

Any task in the expert system can be independently set to manual or autonomous execution. In manual mode, a task selected for execution must first be confirmed by the user. In the manual or confirmation mode, users can approve each step prior to commands being sent to TDAS, however since no plan is generated for the full sequence, the steps must be confirmed one at a time. The HITEX team argued that operators should be presented with a plan for the procedure, and furthermore that they should have access to the underlying logic. Rather than generating a plan, TEXSYS generates steps in a procedure sequentially, making it impossible to generate the kinds of procedure trees operators are used to following in manuals.

TEXSYS performs diagnosis on its own. If the operator believes a certain fault state exists, he would need to issue corrective action from HITEX without waiting for the expert system. Some degree of critiquing function could be implemented using the 'multiple worlds' that KEE provides. However, the truth maintenance system, apparently does not support a thorough critiquing capability.

Ideally TEXSYS would be able to monitor operator commands and issue warnings if it detected a command that would put the thermal bus in an unsafe state.

Intervention and Take Over

If TEXSYS should fail there is an emergency command path directly from HITEX to TDAS so that the operator can safe the bus.

3.4 Supporting User Interface Capabilities

Workspace

HITEX has two monitors, one is a color graphics screen that displays information on the current state of the thermal bus system; this data comes directly from the TDAS. The other is a black and white screen that provides information on the TEXSYS. A mouse and a keyboard is shared between the two monitors.

Tiled window displays are used on both screens. Users can select a particular screen configuration from a fixed menu of options or use the default configuration. These options are different for the graphics screen and for the expert system screen. For example the graphics screen can be configured to be full screen; subdivided into half screens or subdivided into quadrants. The configuration range for the expert system screen is from one large window to six small windows. Users also select what information should be displayed in each tile (window).

The top band of both screens has fixed displays; on the right hand portion of each band is a palette of icons representing local display functions. See figure 3-1 for a view of the overall workspace.

Information and Presentation

The graphics screen has six displays, which are discussed briefly in this section.

1) Global System Parameters Display

   The following parameters are always available (and automatically updated) at the top left of the screen:

   - Setpoint (actual/commanded)
   - Total heatload
   - Subcooling
   - End-to-end delta pressure
   - RFMD voltage
   - Accumulator position

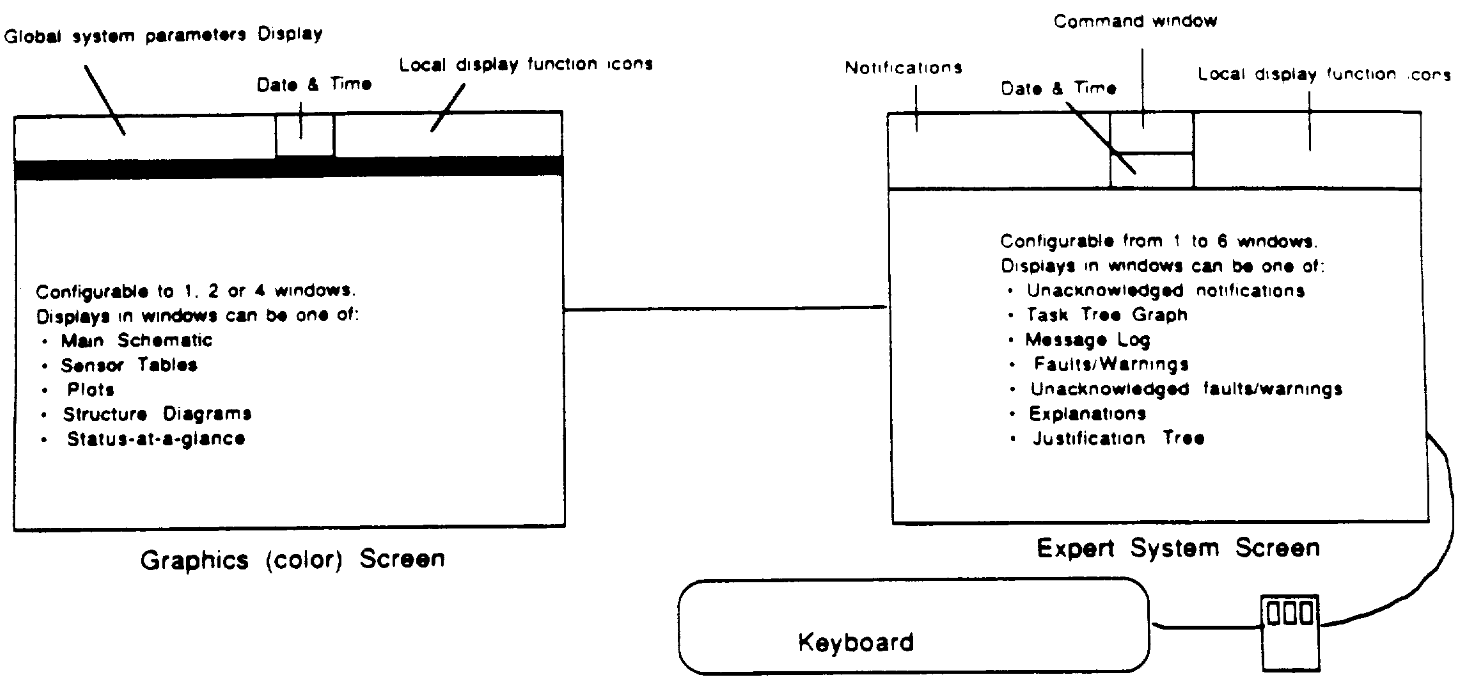


Figure 3-1. HITEX Workspace Design

2) Main Schematic

A mimic of the thermal bus system (from engineering drawings) is shown. Components, sensors, valves and flow lines on the bus are shown as are some other facility structures (e.g. heater carts.) See figure 3-2. The color convention used is as follows:

- Yellow
  Vapor channel
- Orange
  2-phase channel
- Blue
  Warm liquid channel
- Dark blue
  Cold liquid channel
- Purple
  Vent line
- Gray
  Facilities structures, unmodeled components
- Red
  Anomaly
- White
  Major components, isolation valves

All major components (evaporators, condensers, accumulators, RFMD), the BPRV, sensors, and computer-controlled valves are mousable. Mousing on components causes a menu to pop up that offers choices allowing the user to:

- Get a statement about the component's status
- Get a detailed picture of the component
- Remove or display surrounding sensors

Mousing on a sensor presents a menu that allows the user to:

- Get a statement about the sensor's status
- Show the units of measure of the sensor (n.b. these are not displayed by default because of limited screen space)
- Turn the sensor on/off

Mousing on sensor tags, which indicate the location of the sensors, causes the display state of the sensor to toggle between displayed and undisplayed.

Mousing on an electromagnetic isolation valve presents a menu that allows the user to:

- Get a statement about the valve's status
- Open/close valves

For some objects such as certain valves, there is no qualitative information. Mousing on one of these will produce a message saying there is no menu for that item.

3) Sensor Tables

This quarter window display is a list of selected sensor names, a short description, their numeric values and engineering units. The user may create a new sensor table or load one already defined. The table may contain up to 15 sensors without scrolling. This display allows quick viewing of certain sensors that always need to be monitored during, for example, startup or shutdown.

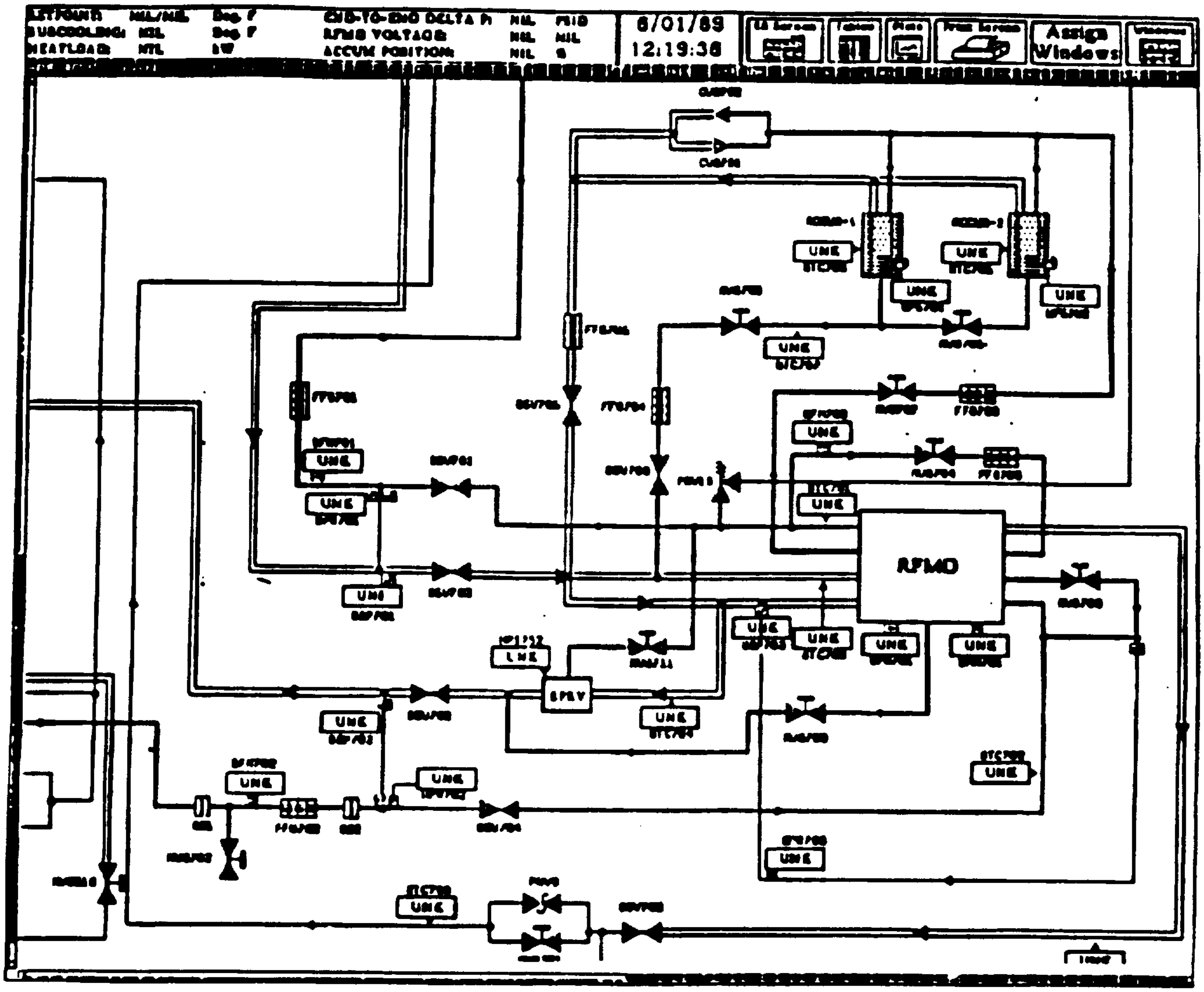


Figure 3-2. HITEX Color Graphics Screen at Startup

4) Plotting

This quarter window display presents a plot of selected sensors (up to five) against a horizontal time axis. The user needs to specify starting time and duration time. Real time data can be plotted as well. Each sensor is color-coded and off-nominal ranges are highlighted for real-time data. Plots can be created and saved, reloaded, activated, and deactivated (the latter were included to conserve processing resources if real-time plots are not currently required). Grid lines are available for clarity. When more than one sensor is plotted in one window, the user may specify two vertical axis ranges to plot against.

Since quarter windows are used for plots, up to four plot windows could potentially be present on the screen at once.

No real-time data will be plotted when TDAS is not communicating with HITEX.

5) Structure Diagrams

These displays are detailed schematics of either condensers or evaporators. One component or a group of components may be displayed. There is dynamic sensor updating. The same color convention is used as in the main schematic. Nothing is mousable.

6) Status-At-A-Glance

This quarter window displays key sensor information in a topological layout. Components are shown as boxes and appear in approximately the same location as in the main schematic. English labels appear under them and are color coded to match the flow lines on the main schematic. Also, sensor readings are located near the component boxes and are updated dynamically. Nothing is mousable.

The following eight displays are found on the expert system screen. See figure 3-3 for a sample screen.

1) Notifications Window

This display is in view at all times because it is where all critical communication with the user occurs. Notifications are posted with a time and date. When notifications are posted, an intermittent alarm sounds. Operator response of some sort (simple acknowledgement, confirmation, choice, key-in) is required for all notifications. Notifications occur under the following conditions:

- Acknowledgement of information or operator action must be taken before expert system may perform some task
- Manual task must first be approved by the operator
- For some tasks the expert system requires the operator to select some choice
- The operator must enter a numeric value
- HITEX specific warning (e.g. communications problems between the expert system or TDAS)

2) Unacknowledged Notifications

Those notifications that have been unacknowledged appear in this window.

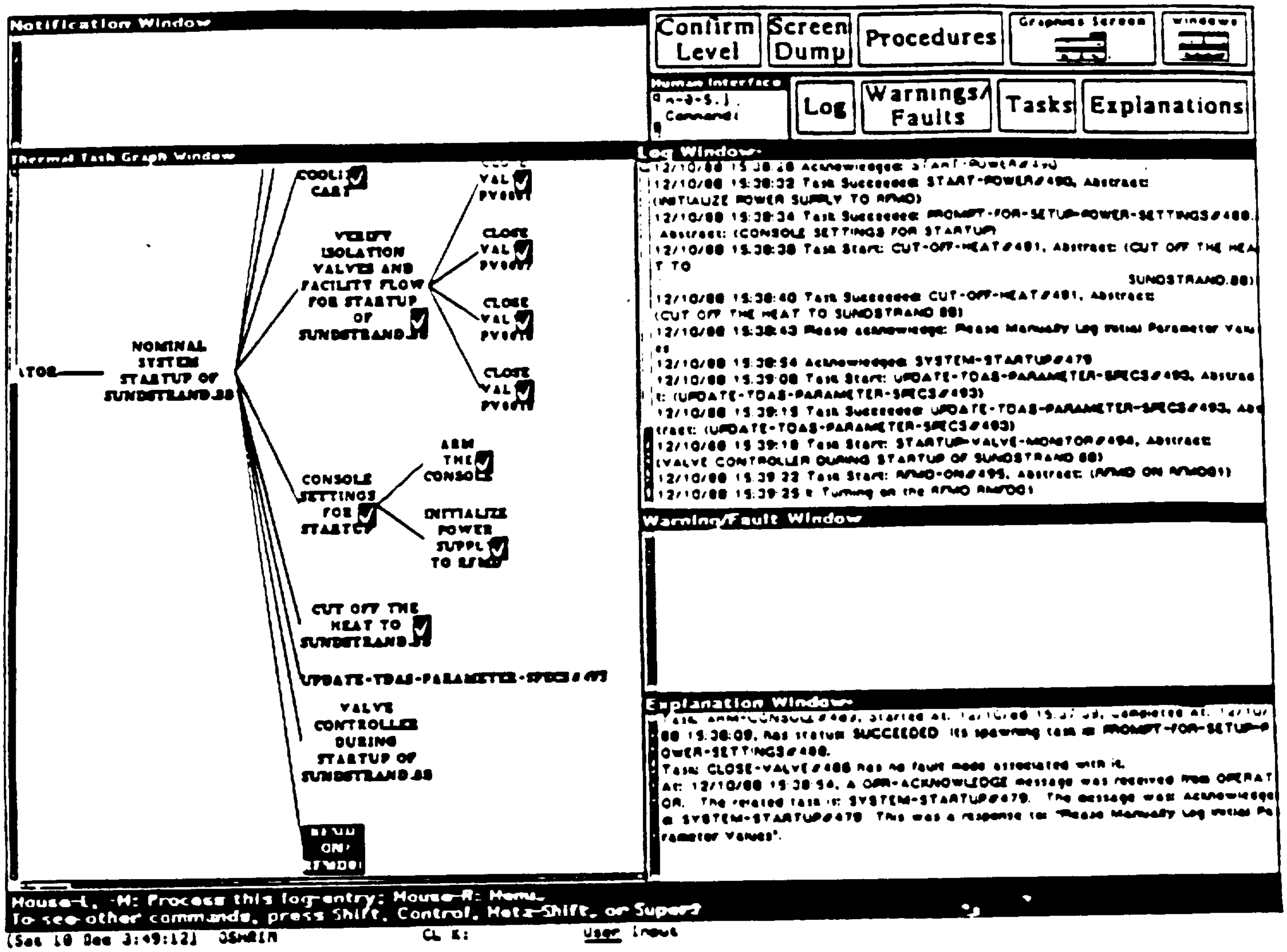


Figure 3-3. Default Window Configuration - E. S. Screen

3) Task Tree Graph

This tree depicts the thermal tasks and subtasks that were performed by the expert system as a result of its reasoning. The tasks and subtasks are shown as nodes and organized hierarchically. The most recently added node is shown in reverse video. Completed successful tasks are indicated by a check; completed failed tasks are indicated by an "X". Serial tasks are indicated by the solid lines leading into them; separate tasks have dashed lines. Each node in the tree is mousable. When selected, a menu allows the user to:

- View a task description
- View a task's status
- Abort a task
- Succeed a task (i.e. terminate it and declare it successfully completed. The interface will only allow this if it is legal; if it is not, an explanations window message will appear.)
- Override a task's waiting time

4) Message Log

The message log keeps a permanent record of all messages into and out of HITEX. Each entry also notes the date and time. The log is updated whether the window is displayed or not. All entries are mousable to display additional information as in other windows.

5) Faults/Warnings Display

This window displays messages about anomalous conditions on the thermal bus, according to the expert system. A beeping accompanies these messages. These messages need to be acknowledged by responding to the message. The beeping may be silenced by clicking on the message to acknowledge it individually or via a Mass Acknowledge icon. When the condition no longer exists, the message disappears from this window.

Each alert message is categorized as either a warning, an alarm or an emergency, depending on its severity. Faults correspond to the determination in the expert system of a diagnosis, or cause for a set of anomalous symptoms. Fault messages include the name of the fault and the name of the task(s) that will attempt recovery.

6) Unacknowledged Warnings/Faults

All warnings and faults are written to this window as they occur; they remain here even when the condition clears up and the message disappears from the Faults and Warnings Window. The console beeps until the message is acknowledged individually or via the Mass Acknowledge icon.

7) Explanations Display

This is the additional information that is obtained via menu by selecting a fault, task or notification. The explanations themselves are mousable as well.

8) Justification Tree

This displays a trace of the conditions that were satisfied when the expert system made its fault diagnosis. Justifications can be accessed by selecting the fault message and then the appropriate menu item. The nodes on this tree are not mousable. Symbolic scrolling is available.

Support for Interaction

1) Modes

The expert system can operate in fully automatic or fully manual mode or in the continuum between these extremes. The mode determines the level of autonomy at which the expert system initiates tasks. These tasks can have an operator-confirmation attached to them or not. The operator can add (and remove) confirmability to tasks at his own discretion via the Confirm Level icon.

2) Interaction features

Mouse-selectable menus are used to present choices to the operator. There is one 3-buttoned mouse for both monitors; the user toggles between monitors by clicking on a box at top of the screen.

Window configurations are selected from a menu of images representing the possible configurations. Whenever a display is selected, a window must be available for it. The user typically selects the display label and then selects the window in which it should appear. Most displays are designed for a quarter window, and will appear best in one, however most do not require this size.

Many windows are scrollable. The scroll bars functionality native to the Symbolics computer is used. Some windows also have symbolic scrolling via the middle mouse button, however this functionality is slow.

Full screen or individual window hardcopies are available by selecting an appropriate menu item.

Feedback to an input is given by a cursor change (hourglass). Also, "run bars" at the bottom of the expert system screen indicate the type of system processing occurring, e.g. garbage collection indicated by the first bar, system waiting for disk indicated by second bar, etc.

Many thermal tasks may be initiated by the operator on the expert system screen. The operator initiates the task by selecting from a menu in the Procedures icon. A query confirms with the operator that it was not selected inadvertently. A left click confirms; middle or right aborts.

3.5 Design Process and Final Notes

A formal software requirements document was written based on input from domain experts, NASA software standards, and NASA display and control requirements. Early on the "spiral" model of development was chosen in which successive prototypes are shown to users and their feedback is incorporated into successive designs. (For more detail see Hack and DiFilippo, 1990.) It was found, however, that the redesign of the interface and interaction functionality as aspects of the intelligent system or monitored process changed required significant effort; one main reason may have been the lack of a model-based interface and interaction functionality.

The project was comprised of team members from several different sites: NASA-Ames, NASA-JSC, Rockwell, Boeing, and Sterling Federal Systems. The variety of expertise led to some communication problems and differences of opinion concerning the values of certain efforts, such as documentation. As a result, the task of documentation was assigned primarily to one person towards the end of the project. Also "the physical separation of the knowledge engineers and the domain experts slowed and complicated the development process." (Hack and DiFilippo, 1990).

Access to the user population for this system was very limited, however the HITEX team planned to involve expert users from the start of the project. They had weekly teleconferences or videoconferences between ARC and JSC. Many of the problems were found and solved during the JSC engineers visits at ARC, and during the ARC developers' stay at JSC. In situations where early versions of HITEX were shown to users, the team found that the low fidelity of the stand-alone prototype inhibited meaningful results.

Trained engineers at JSC mainly used the graphics (Thermal Bus) screen to validate TEXSYS conclusions. The Expert System Screen was used more as a developer's tool for validating reasoning. According to Remington and Shafto (1990), "We felt the expert system should aid the operator. The controlling viewpoint, however, was that the expert system should demonstrate its capacity for autonomous control. Thus, it was often impossible to convey the desirability of specific interface features, or to prevail if a desired feature placed increased demands on expert system development."

One of the problems with TEXSYS was that using KEE's truth maintenance capability slowed the system considerably particularly after a few hours of operation. "Though TEXSYS was nominally model-based, it could not handle the computational demands of diagnosis combined with state extrapolation." (Remington and Shafto, 1990). See figure 3-4 for a summary of "Lessons Learned" from this project (JSC/ARC, 1989).

3.6 Summary of Issues

- Functional-Based Displays

  The "Status at a Glance" display shows the recognized need for a function-oriented display that would provide the operator with a quick, overview picture of the system's state. However, the particular display used is unlikely to meet this need because the individual parameter values are not integrated with respect to a model of system functioning. Furthermore, they are digitally displayed without data on limits and targets. See section 5.4 (Physical Topology Schematics) in Volume 1 (Malin et al., 1991) for further discussion.

- Message Lists

  Some types of data are displayed as textual message lists. The problems of message lists for temporal organization of sequences of events are discussed in section 5.3 (Message Lists and Timeline Displays) in Volume 1 (Malin et al., 1991).

  An example of a representation that goes beyond message lists is the task tree display observed in this case. It organizes event information in a graphical, hierarchical manner.

- Physical Topology Schematic Displays

  There are several schematic displays which include pan and zoom. It should be noted that this kind of graphic form, coupled with an overemphasis on the digital display of state variables, may not sufficiently highlight events and anomalies in the monitored process. See section 5.4 (Physical Topology Schematics) in Volume 1 (Malin et al., 1991) for further discussion.

SADP TCS - ADVANCED AUTOMATION DEMONSTRATION OF SPACE STATION FREEDOM THERMAL CONTROL SYSTEM

SADP LESSONS LEARNED

MOST IMPORTANT

- Identify the user at start and have him closely involved continuously.

DOMAIN EXPERTISE

- Applications should be well developed so that detailed FDIR procedures are available to begin knowledge extraction early.
- Domain experts should periodically review the knowledge base in detail to gain insight into the expert system design.

KNOWLEDGE ENGINEERING

- Knowledge engineering requires close working relationships and is difficult to accomplish when physically separated.

EXPERT SYSTEM DEVELOPMENT

- Realtime expert systems should utilize software development tools compatible with the realtime applications requirements.
- Formalism of software development practices and schedule firmness should correlate with project technical risk levels.

Figure 3-4. Systems Autonomy Demonstration Project (SADP) Lessons Learned

- Workspace Navigation
  Overall, the number and flexibility of windows and alternative views has the potential to create navigational difficulties and extra data management burdens. See section 5.2 (Workspace: Proliferation of Windows) in Volume 1 (Malin et al., 1991).

3.7 Review Method

This case report was based exclusively on the documents listed below.

3.8 Case Data Sources

Dorighi, N., G. Davis, R. Frainier, and A. Oshrin (June, 1989), *Human Interface to the Thermal Expert System (HITEX), User Manual*, Systems Autonomy Demonstration Project, NASA Ames Research Center.

Hack, E. C. and D. M. DiFilippo (March, 1990), "Demonstrating Artificial Intelligence for Space Systems Integration and Project Management Issues", IEEE Conference on Artificial Intelligence Applications, Houston, TX: Lockheed Engineering and Sciences Company.

JSC/ARC (August,1989), "Advanced Automation Demonstration of Space Station Freedom Thermal Control System", briefing about Systems Autonomy Demonstration Project.

Malin, J. T., D. L.Schreckenghost, D. D. Woods, S. S. Potter, L. Johannesen, M. Holloway, and K. D. Forbus (September, 1991), *Making Intelligent Systems Team Players: Case Studies and Design Issues, Volume 1. Human-Computer Interaction Design*, NASA Technical Memo, Houston, TX: NASA - Johnson Space Center.

Remington, R. W. and M. G. Shafto (April, 1990), "Building Human Interfaces to Fault Diagnostic Expert Systems I: Designing the Human Interface to Support Cooperative Fault Diagnosis", Conference on Human Factors in Computing Systems, CHI '90 Workshop: Computer Human Interaction in Aerospace Systems, Moffett Field, CA: NASA - Ames Research Center.

Shafto, M. G. and R. W. Remington (April, 1990), "Building Human Interfaces to Fault Diagnostic Expert Systems II: Interface Development for a Complex, Real-Time System", Conference on Human Factors in Computing Systems, CHI '90 Workshop: Computer Human Interaction in Aerospace Systems, Moffett Field, CA: NASA - Ames Research Center.

Section 4
Spacecraft Health Automated Reasoning Prototype (SHARP)

4.1 System Description

SHARP is a ground-based prototype system designed to monitor the health and status of multi-mission spacecraft and the ground data systems operations. Its expert system performs real time anomaly detection and diagnosis. SHARP compares the systems' behavior based on real-time data with the data it has in its rule base concerning how these systems should be operating. If expected behavior does not correspond to actual, SHARP informs the operator that an alarm condition exists and lists the possible causes and suggests actions to take.

SHARP basically attempts to provide telecommunications personnel with an information environment to better understand how the telecommunications link is functioning.

The goals of this effort are to:

- Eliminate the tedious manual processing and analysis required to assess spacecraft and telecommunications status
- Provide more efficient and reliable identification of problems. (Currently the diagnosis and recovery from failure depends upon the crucial skills of experts who are in high demand.)

The monitoring and troubleshooting of the telecommunications system on Voyager 2 and the telecommunications link analysis during its Neptune encounter served as SHARP's initial demonstration environment (August 1989).

Monitored Process

Telecommunications downlink data, 'channelized data', about spacecraft parameters to three complexes of antennas (Deep Space Stations, or DSSs) spread around the world. Each unmanned spacecraft must be monitored continuously. During peak periods of the Voyager 2 mission, up to 40 real-time operators are required to monitor over a 24-hour, 7 day/week cycle.

Important functions in the analysis of the telecommunications link are to: 1) numerically estimate the performance of the telecommunications subsystems and link, 2) monitor the telecommunications in real time, 3) detect problems 4) diagnose and recover from these problems. The following types of data and information are needed in order to perform these functions.

Data and Information Used and Current Practice

Predicts are numerical predictions for spacecraft and DSS station parameters that impact the performance of the telecommunications link, such as, signal-to-noise ratio or antenna elevation angle. There are four kinds of predicts. In the present telecommunication environment much of the predict calculation is done manually; it is both tedious and time-consuming.

The Integrated Sequence of Events (ISOE) is a time-ordered sequence of spacecraft and DSS activity. ISOE data are used in predict calculation, alarm determinations and anomaly diagnosis. In the current telecommunication environment the appropriate ISOE in hardcopy form is visually scanned by the operator and telecommunication events are manually highlighted so that the relevant telecommunication activity can be monitored. Modifications are made to the original listing by issuing handwritten correction sheets. The ISOE's tend to be extensive, and it is possible for an event to be embedded among several pages of another subsystem's events. This method can make it difficult to spot events, particularly during peak activity periods. In monitoring critical events, operators tend to rely upon the Spacecraft Flight Operations Schedule (SFOS) which is an unofficial graphical sequence of events. Problems can occur when operators do not refer to the latest activity modifications.

The telemetry data from spacecraft, tracking stations and other systems is separated into channels, each containing information regarding a single system, subsystem or component. This channelized data specifies the values of hundreds of spacecraft engineering status parameters ('engineering parameters') and station performance parameters ('monitor parameters'). Currently, the channels are plotted on black and white video terminals and are visually scanned to make sure they are within their prespecified limits.

The channelized data is compared with the alarm limits. The alarm limit values vary depending on the status of several parameters, such as the state of the spacecraft instruments and spacecraft events. Currently, the these limits are changed manually in real time. This must be done often and typically operators select a wide threshold so that the entire range of parameter conditions is reflected; however broadened alarm limits increases the risk of undetected anomalies.

Man-Machine System

SHARP was developed to eventually support the workload of the Space Flight Operations Center (SFOC), a multi-mission flight operations team that was established to operate all unmanned spacecraft. The goal is for the SFOC team to operate all of the spacecraft using the SHARP technology as a centralized information access and diagnostic tool.

Development and Testing Environment

SHARP exists on one workstation. SHARP is implemented in Common Lisp on a Symbolics™ 3650 color LISP machine. It uses the STAR*TOOL expert system building language and environment (developed at JPL). Some of the functions that STAR*TOOL provides are: a blackboard architecture for communication within heterogeneous multi-process environments, a parallel control structure, and truth maintenance.

4.2 Intelligent System and Functions

Real-time Anomaly Detection and Diagnosis

SHARP receives real time data on how the spacecraft and DSS is performing. It detects anomalous data values and performs diagnosis of these faults. The data coming from various channels is checked against each channel's alarm limits. Alarm limits for the spacecraft are dependent on the spacecraft's state; alarm limits for ground stations are fixed.

If an alarm occurs (fault is detected), the diagnostic system is activated. The diagnostic system consists of a hierarchical executive diagnostician and several mini-experts. Each mini-expert handles the local diagnosis of a specific class of faults (e.g. particular channels in alarm, loss of telemetry). Some mini-experts ("cooperating mini-experts") search beyond their local area

---

™ Symbolics is a trademark of Symbolics, Inc.

when a fault cannot be identified by examining a single fault class. The executive diagnostician, whose rules execute in pseudo-parallel, then integrates the input from the mini-experts to propose one or more fault hypotheses. Barring a failure of the diagnostician or a mini-expert, the system is capable of multiple fault diagnosis. Truth maintenance capability monitors for violations of logical consistency and provides functions for temporal reasoning in multiple fault diagnosis.

4.3 Human-Intelligent System Interaction Functions

Assessment

1) Alarm Situations and Diagnosis

   Users are notified of alarm warnings no matter what display is currently on the screen. The user may specify what classes of alarms should or should not be presented to him, and may also specify that the alarm windows automatically close after a certain time period has elapsed. In this same window is presented the results of the diagnosis. If more than one hypothesis ("explanation") is generated, they are listed in order of plausibility, and, along with each, is a suggested action to take. Figure 4-1 shows an example of such an alarm warning. Each new alarm is recorded in an Alarm History display and also an alarm meter is created and displayed in a separate Alarm Meters display. Users are informed of the number of current alarms by an Alarm Status box that is always present on the upper right of the screen.

2) Status Information

   In order to monitor the spacecraft and telecommunications link the operator needs several types of data and information. As mentioned earlier, operators currently must integrate much information, perform calculations manually, and rely upon several different hardcopy forms for information. SHARP provides the following functions that attempt to update current practices:

   - Performs predict calculation in real time based on the raw predicts, and presents it to the operator on a special display
   - Keeps track of ISOEs, allowing the operator to search, and review summaries of selected events. SHARP's ISOE database may be updated by users in order to reflect the real time commands sent to the spacecraft
   - Allows the operator to quickly identify available stations in case these are needed
   - Allows for customized display of channelized data plots
   - Determines alarm limits dynamically and provides for on-line viewing and editing of alarm limits
   - Provides on-line schematics of the components and subsystems of the telecommunications path
   - Presents a dynamic graphical view of spacecraft motion parameters
   - Provides user with time ranges and explanations of data outages and warns user when and why to expect noisy data

```
Message from the diagnostician at GMT 163 20:22:54;

Hard Alarm: E-025 S/C AGC (Automatic Gain Control) in alarm.

Source: S/C Engineering Data.

Explanation 1: The DSN Exciter Frequency is in alarm.  The wrong ramp
was entered into the Digitally Controlled Oscillator (DCO).

Corrective Action: Advise DSN to restart the DCO with correct frequency offset
and ramp rate.

Explanation 2: The AGC detector has failed.

Corrective Action: Notify SCT personnel.

Explanation 3: S/C Antenna is off point.

Corrective Action: Check Attitude Control data.

More Information: Consult data in the Channelized Data Display
for channels E-025 (S/C AGC), M-777 (Exc Freq), E-074 (Attitude Pitch),
E-181 (Attitude Yaw), and E-189 (Attitude Roll).

Please click any mouse button to acknowledge.
```

Figure 4-1. Alarm Warnings

- Performs a Fast Fourier Transform (FFT) of the DSS conical scanning component to indicate when the antenna is going off point. This allows the operators to detect the problem quickly and contact the station to correct the antenna in order to prevent the loss of data.

Section 4.4 on Supporting User Interface Capabilities describes the different types of data/information presented in the context of the specific displays used.

Collaboration

Fault detection and diagnosis is performed exclusively by the system. The system lists the potential causes for the anomalies and suggests actions to take to respond to alarm conditions.

By automating the process of telecommunications diagnosis, the procedure for responding to an alarm does not require that hundreds of people be notified and put on alert. Part of the current standard procedure is that experts must be consulted even when the situation arises from a known false alarm. "By improving the monitoring process and correcting some of the inaccuracies of the current system, SHARP attempts to produce far fewer false alarms and reduce the mundane procedures required in handling the known common problems. When alarm conditions arise from any monitoring procedure within the SHARP system, such as channels in alarm, link status problems, antenna tracking errors, or attitude alarms, the information is automatically passed to the [SHARP's] diagnostician to determine all possible causes for the anomaly" (Martin et al, 1990).

Besides reviewing the data stored in SHARP, operators may need to change the data stored. The only data that should be modifiable by operators are: alarm limits, ISOEs, and predicts. Temporary changes may be made to alarm limits (to try out hypothetical situations?); in doing so the operator may suppress alarms or set tighter alarm limits for closer scrutiny of a particular event.

Intervention and Takeover

SHARP does not take any control actions of its own.

The operator can make permanent changes to the alarm limits; an operator may need to do this if there are changes in spacecraft or DSS operations or performance. These changes can be written to the data files so they will be available the next time SHARP is started up.

4.4 Supporting User Interface Capabilities

Workspace - Information and Presentation

There is one workstation. A layout of the generic workstation is shown in figure 4-2. The Program Menu lists the twelve screen displays, which appear in the Screen Display Area. The Display-Specific Menu lists options that are relevant to a particular screen display. The Alarm Status window presents the number of active alarms. The Interaction Area allows developers and more experienced users to access the LISP environment to check SHARP status and recover from errors. All the display screens are briefly summarized below. The first three are displays that allow review and change of data. The rest are displays driven by real-time data.

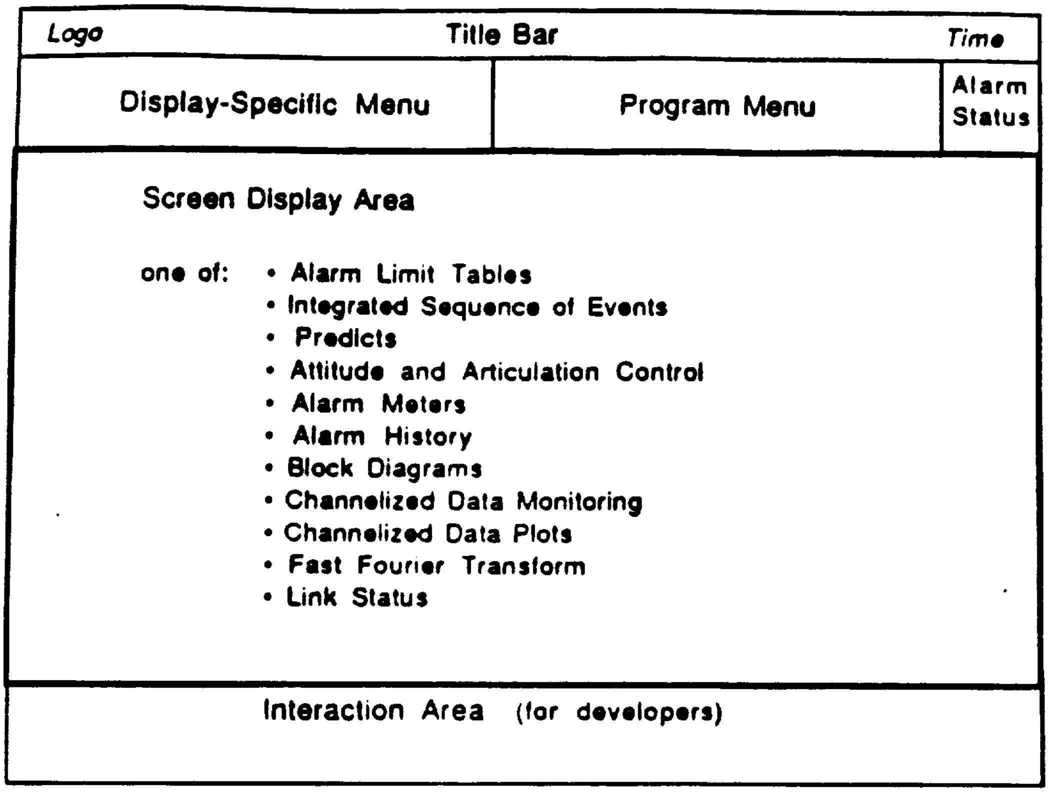


Figure 4-2. SHARP Workspace Design

1) Alarm Limit Tables

This display allows operators to view and edit the list of spacecraft engineering alarm limits (nominal value boundaries), DSS performance limits, ground data system limits, and residual thresholds. The user must select which of these to view. For each type of channel chosen, several types of values are displayed. See figure 4-3.

Temporary changes (such as suppressing alarms or setting tighter alarm limits) are allowed to enable operators to study a particular event. Also, the operator may permanently change the alarm tables by saving them.

2) Integrated Sequence of Events

ISOE information is the schedule of spacecraft activity integrated with corresponding DSS tracking activity. An ISOE, in a tabular form, may be viewed for any user-specified day or week. Users may also request status summaries of any activity. For example, the user may request to see all values of the variable "Data Mode" during a user-specified period of time.

The ISOEs may be edited via menu driven commands. A history of user edits is maintained so the user can verify these changes. See figure 4-4.

3) Predicts

Both tabular and graphical display of prediction data (raw predicts, pass predicts, instantaneous predicts and residuals) for any specified time range is available. The user may also request to see a color-coded graph of the DSS that are in view of the spacecraft during a particular time frame. This allows for rapid identification of available DSSs. See figure 4-5.

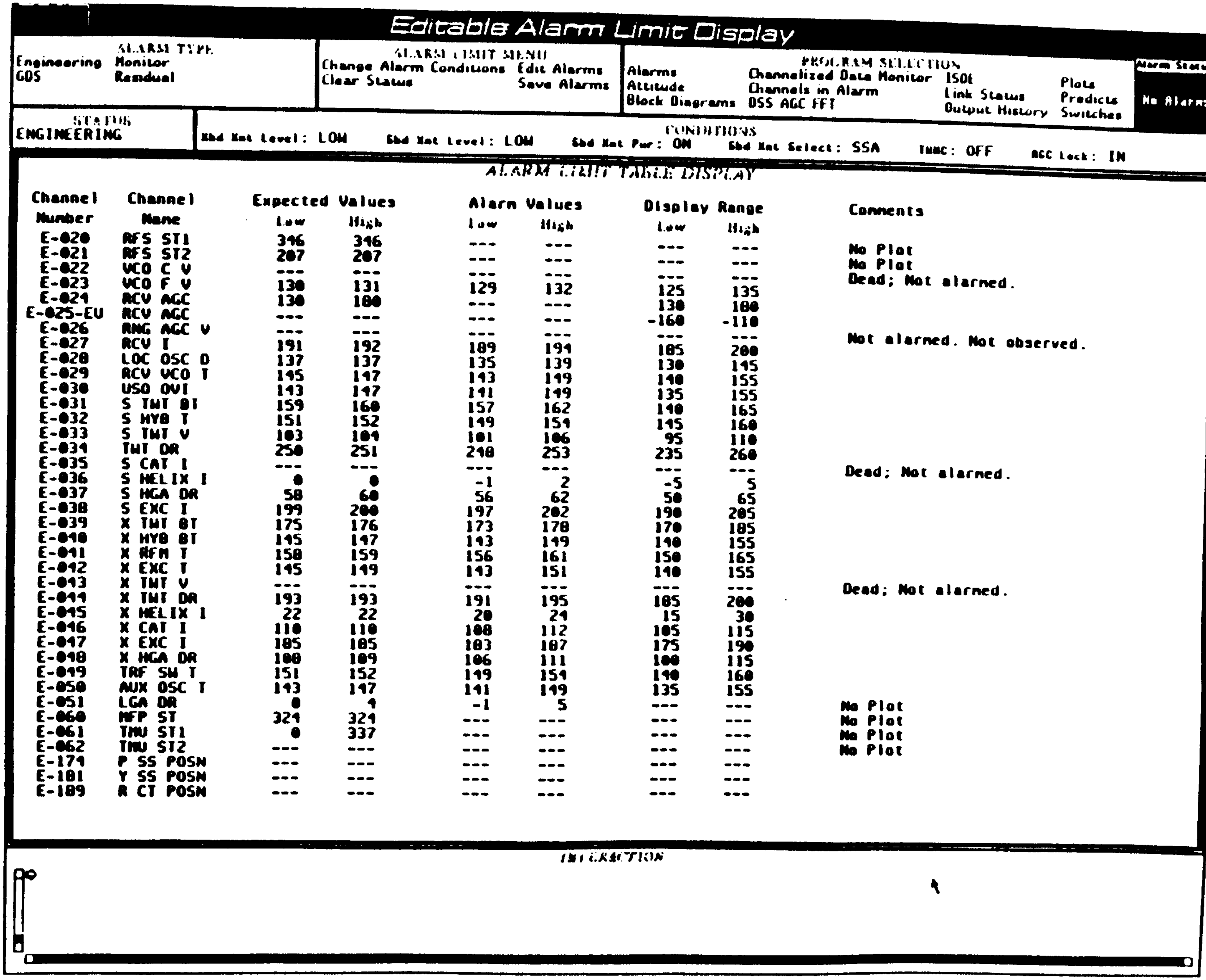
4) Attitude and Articulation Control

This display provides an integrated graphical representation of spacecraft attitude and articulation; currently users must examine individual plots of spacecraft pitch, yaw and roll movements.

The spacecraft is represented by an large crosshair icon. A Maltese cross around the spacecraft icon represents the roll deadband box. The limit cycle box around the spacecraft represents the deadband limits. (This box changes size and shape when pitch and yaw deadbands change.) Yaw and pitch value changes are represented as trailing vectors. This allows for visualization of the spacecraft's movement through time. A color change to red in the icon representing the spacecraft indicates an alarm condition in any of the attitude parameters. The relevant pitch, yaw, or roll information is also highlighted red. See figure 4-6.

5) Alarm Meters

This display shows only channels in alarm. These are shown as meters; a pointer indicates where the channel value is relative to the alarm limits. A label above the pointer specifies the numerical value and the time of the last datum on the channel, and whether the alarm limits on the channel have changed since the datum was reported. Simple numerical alarm limits have regions colored red, yellow and green to indicate the range of hard, and soft alarms and nominal behavior. All other (more complicated) meters are colored blue. See figure 4-7.



Channel Number	Channel Name	Expected Values Low	Expected Values High	Alarm Values Low	Alarm Values High	Display Range Low	Display Range High	Comments
E-020	RFS ST1	346	346	---	---	---	---	No Plot
E-021	RFS ST2	207	207	---	---	---	---	No Plot
E-022	VCO C V	---	---	---	---	---	---	Dead; Not alarmed.
E-023	VCO F V	130	131	129	132	125	135	
E-024	RCV AGC	130	100	---	---	130	100	
E-025-EU	RCV AGC	---	---	---	---	-160	-110	
E-026	RNG AGC V	---	---	---	---	---	---	Not alarmed. Not observed.
E-027	RCV I	191	192	189	194	185	200	
E-028	LOC OSC D	137	137	135	139	130	145	
E-029	RCV VCO T	145	147	143	149	140	155	
E-030	USO OVI	143	147	141	149	135	155	
E-031	S TWT BT	159	160	157	162	140	165	
E-032	S HYB T	151	152	149	154	145	160	
E-033	S TWT V	103	104	101	106	95	110	
E-034	TWT DR	250	251	248	253	235	260	
E-035	S CAT I	---	---	---	---	---	---	Dead; Not alarmed.
E-036	S HELIX I	0	0	-1	2	-5	5	
E-037	S HGA DR	58	60	56	62	50	65	
E-038	S EXC I	199	200	197	202	190	205	
E-039	X TWT BT	175	176	173	178	170	185	
E-040	X HYB BT	145	147	143	149	140	155	
E-041	X RFM T	158	159	156	161	150	165	
E-042	X EXC T	145	149	143	151	140	155	
E-043	X TWT V	---	---	---	---	---	---	Dead; Not alarmed.
E-044	X TWT DR	193	193	191	195	185	200	
E-045	X HELIX I	22	22	20	24	15	30	
E-046	X CAT I	110	110	108	112	105	115	
E-047	X EXC I	185	185	183	187	175	190	
E-048	X HGA DR	108	109	106	111	100	115	
E-049	TRF SW T	151	152	149	154	140	160	
E-050	AUX OSC T	143	147	141	149	135	155	
E-051	LGA DR	0	4	-1	5	---	---	No Plot
E-060	MFP ST	324	324	---	---	---	---	No Plot
E-061	TMU ST1	0	337	---	---	---	---	No Plot
E-062	TMU ST2	---	---	---	---	---	---	No Plot
E-174	P SS POSN	---	---	---	---	---	---	
E-181	Y SS POSN	---	---	---	---	---	---	
E-189	R CT POSN	---	---	---	---	---	---	

Figure 4-3. Alarm Limit Display

C-3

ORIGINAL PAGE IS
OF POOR QUALITY

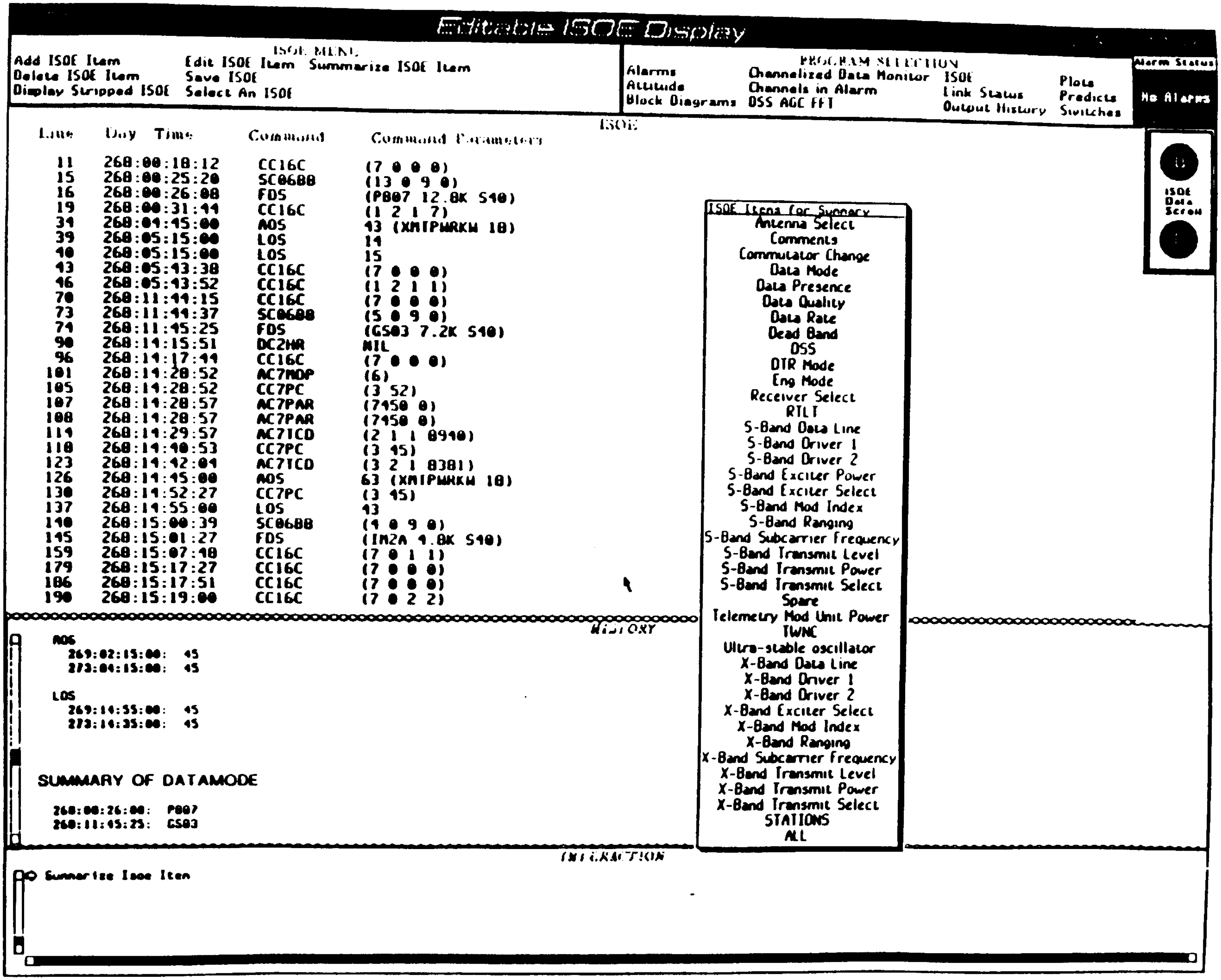


Figure 4-4. Integrated Sequence of Events

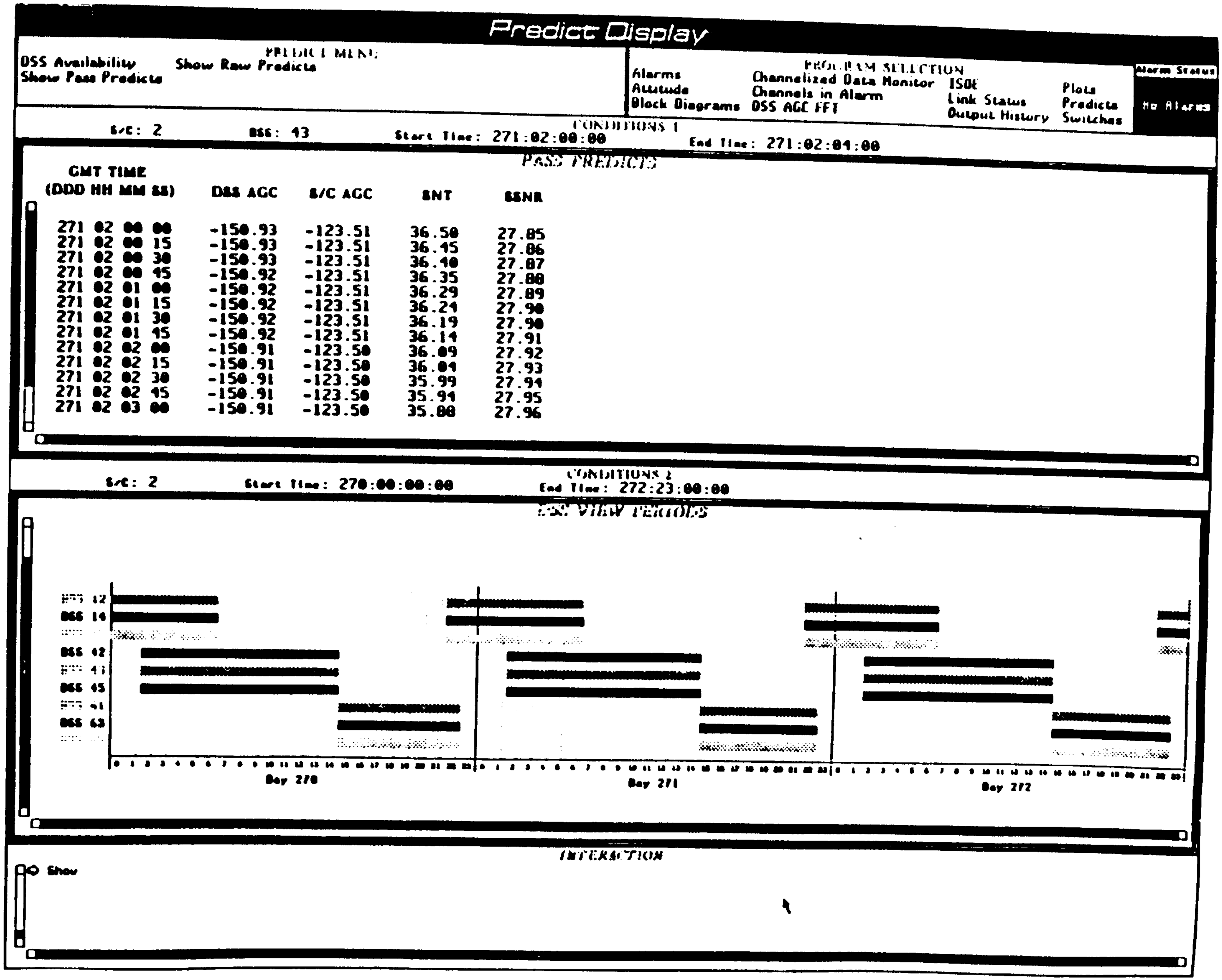


Figure 4-5. Predicts Display

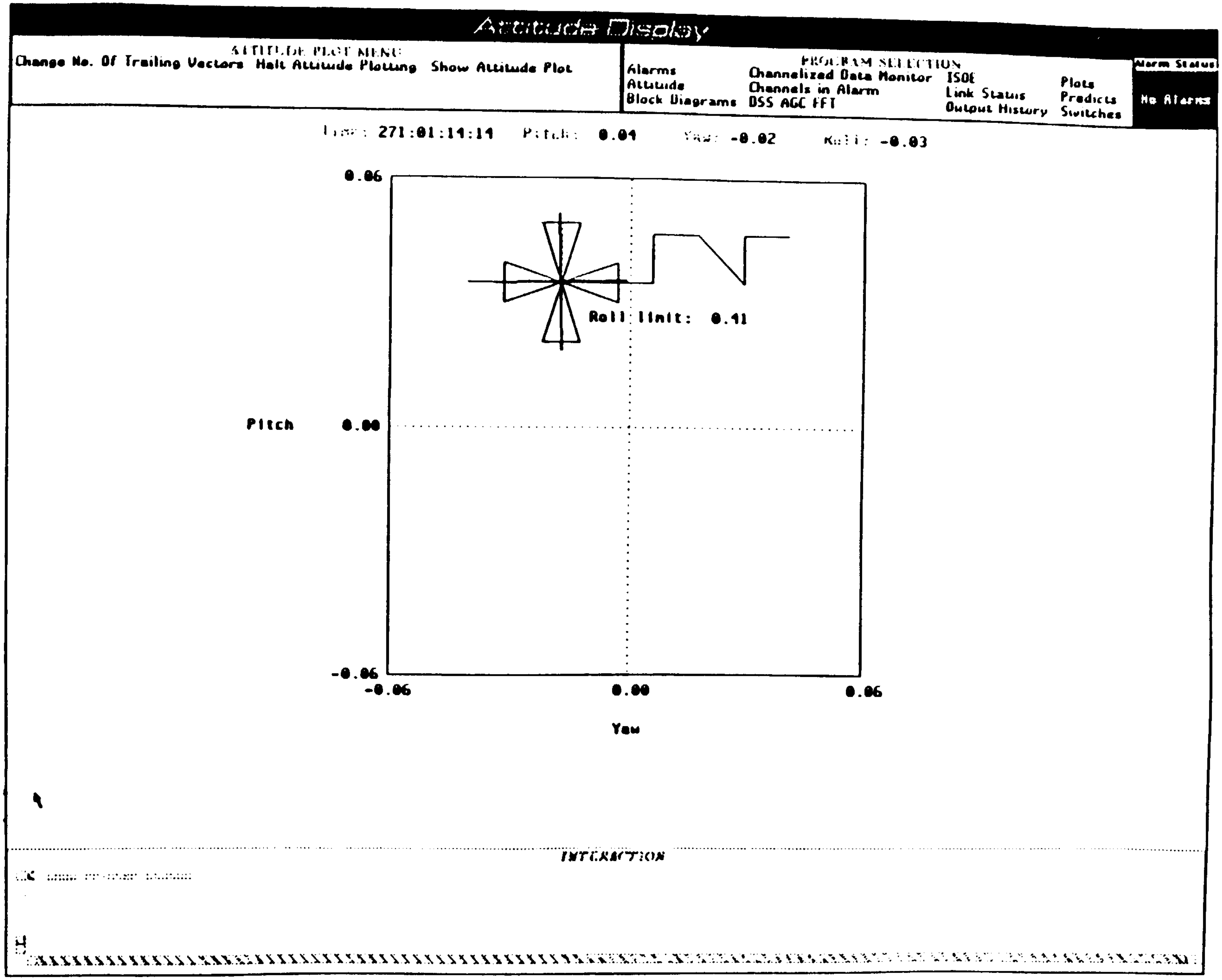


Figure 4-6. Attitude and Articulation Control Subsystem Display

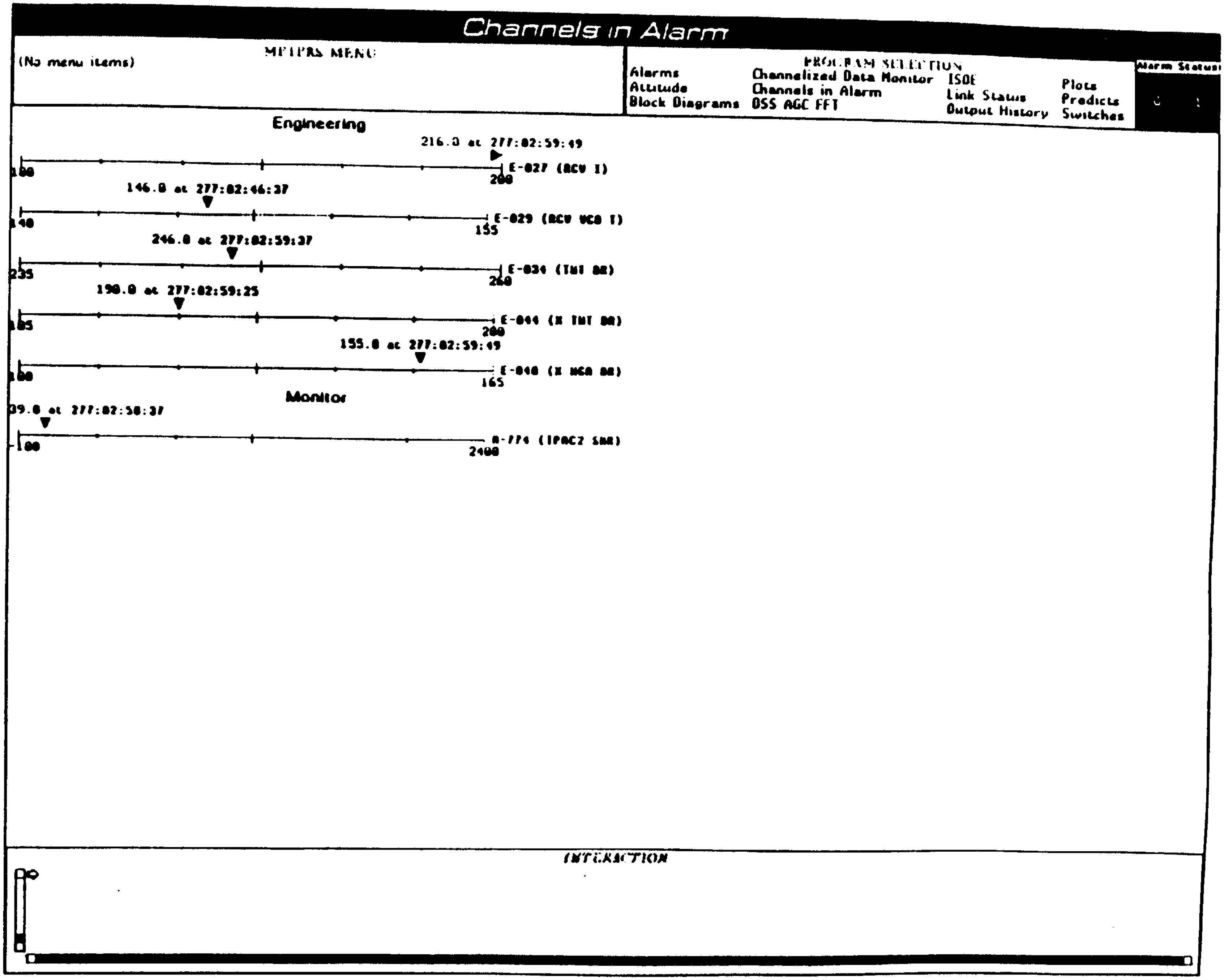


Figure 4-7. Alarms Meters

6) Alarm History

This display allows the user to scroll through a list of the alarm messages that have been generated. Information on how many of each type of message (e.g. alarm, diagnostic) has been generated is also given.

7) Alarm Warnings

This is actually a pop-up window that appears whenever an alarm occurs regardless of the context (unless the user has specified that alarm warnings should be not displayed, or that only one class of alarm warnings be displayed. It is also possible to set a time-out period so that the window automatically closes when it elapses.) This window specifies the alarm type (either soft or hard) and possible explanations for its occurrence along with suggested corrective actions. This window closes upon acknowledgement (mouse click) by the user.

8) Block Diagrams

This display depicts the communication path from the spacecraft through a DSS and Ground Communications Facility to the Mission Control and Computing Center at JPL and the final destination of the Test and Telemetry System computers. It allows operators to view the components and subsystems of the telecommunications path in a schematic form. Different levels of detail are available as the user moves through the block diagram (via mouse clicks). Block diagrams are dynamically updated in real time to reflect the current system configuration (green=ON, white=OFF), connections (switches move), and trouble status (red=subsystems in alarm.) See figure 4-8.

9) Channelized Data Monitoring

This module allows the developer/maintainer to watch and debug SHARP in the context of its data sources. It presents status icons in several formats including: raw number, tables, graphs, meters and textual displays.

10) Channelized Data Plots

The user can select to view several different channels (generally four to a display page is used as it is what is preferred by operators) against a user-specified time scale and data range. Each plot can be color-coded by the user for easy viewing. When a channels in in alarm it is plotted red (hard alarm) or yellow (soft alarm). The actual alarm limits may be optionally overlaid onto the channel's plot. Each data point is mouse sensitive, and when clicked provides time and numerical value indicators. Real time data can be plotted as it is being received. Past data plots that have been stored can be viewed as well. These displays can be scrolled along the time axis. These plots can also represent information as graphs of actual or derived data vs. time, x/y plots or scatter plots. See figure 4-9.

11) Fast Fourier Transform

This display shows a bar chart of Fast Fourier Transforms of the automatic gain control channel (AGC). See figure 4-10. The FFT indicates when the antenna is going off point. This allows the operators to detect the problem quickly and contact the station to correct the antenna in order to prevent the loss of data.

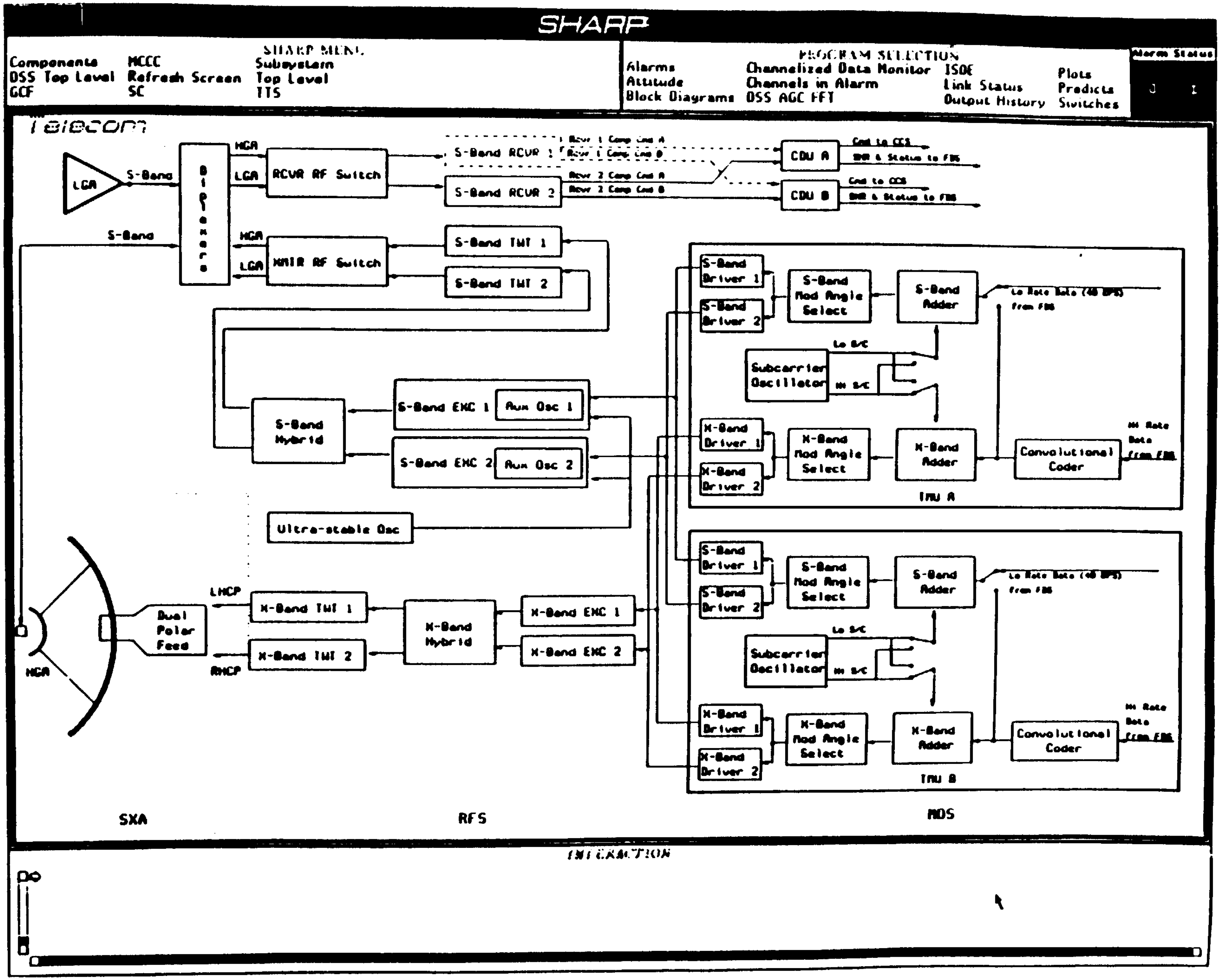


Figure 4-8. Voyager Telecommunications Subsystems Block Diagram

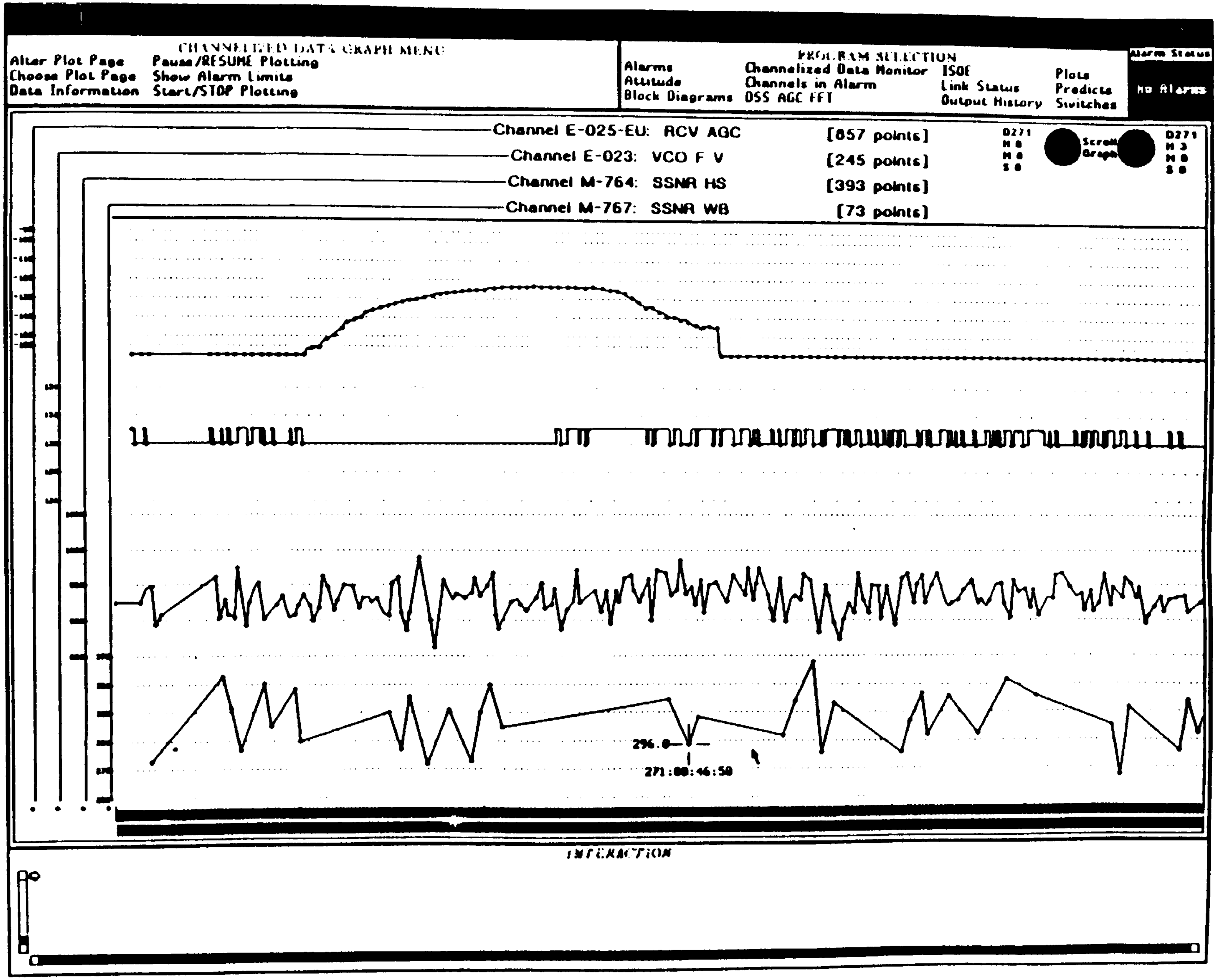


Figure 4-9. Channelized Data Plot

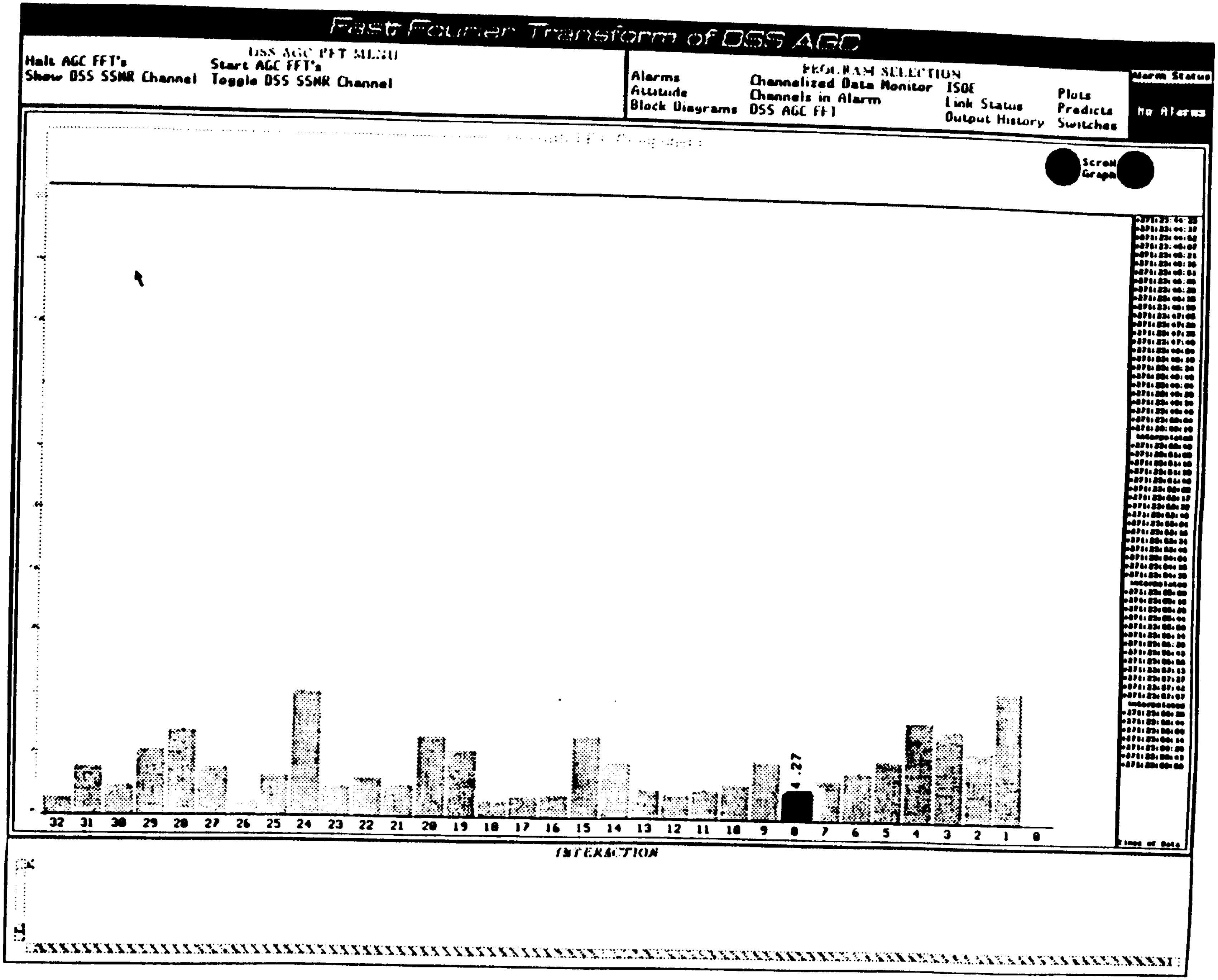


Figure 4-10. FFT Display of DSS AGC

12) Link Status

This display integrates station coverage, spacecraft transmitter power status, data rate, station uplink, projected downlink, data outages, spacecraft-DSS lock status, and spacecraft data quality in one graphical presentation. Time appears on the horizontal axis in hourly (default) increments. The different stations are color-coded. The graph provides valuable time range information, explanations of data outages and can warn the user about when to expect noisy or corrupted data. See figure 4-11.

4.5 Design Process

Initial Observations by End Users

Due to time constraints during the Neptune Encounter, telecommunications operators did not get a chance to evaluate the system as fully as the developers would have liked. However, the reactions that the operators expressed seem to be enthusiasm for its potential. They also mentioned they would have liked a more responsive display interface. The developers point out that "[t]his version of SHARP was not built to be an operational system with minimal bugs; it was built as rapidly as possible to get a prototype running to determine if the ideas motivating the SHARP architecture were correct" (Martin, 1990). The developers feel these are correct and are continuing work on improving their implementation.

4.6 Summary of Issues

- Integration of Related Data

  Relevant data can be integrated in a way that allows for pattern-recognition in situation assessment. This has the potential for better and more mentally economical situation assessment. One example that attempts to do this is the Attitude and Articulation display.

  There are several displays that collect and integrate data based on an operator's perspective. For example, The Integrated Sequence of Events display attempts to do this by presenting the data organized by summaries of spacecraft activity or status summaries of a specified activity. Also, the Alarm Meters display shows only those channels in alarm, and situates the values in a context by providing range and nominal information.

- Message Lists

  There are some plan-oriented timeline displays. Concepts from these plan-oriented timelines may be useful in creating alternatives to textual message lists for sequences of abnormal events.

  Some textual message lists are used (for example, the data in the Alarm Warning Window). The problems of message lists for temporal organization of sequences of events are discussed in section 5.3 (Message Lists and Timeline Displays) in Volume 1 (Malin et al., 1991).

4.7 Study Method

This case report was based heavily on the available reports on SHARP (see below) and to a lesser extent on a brief demo of the system.

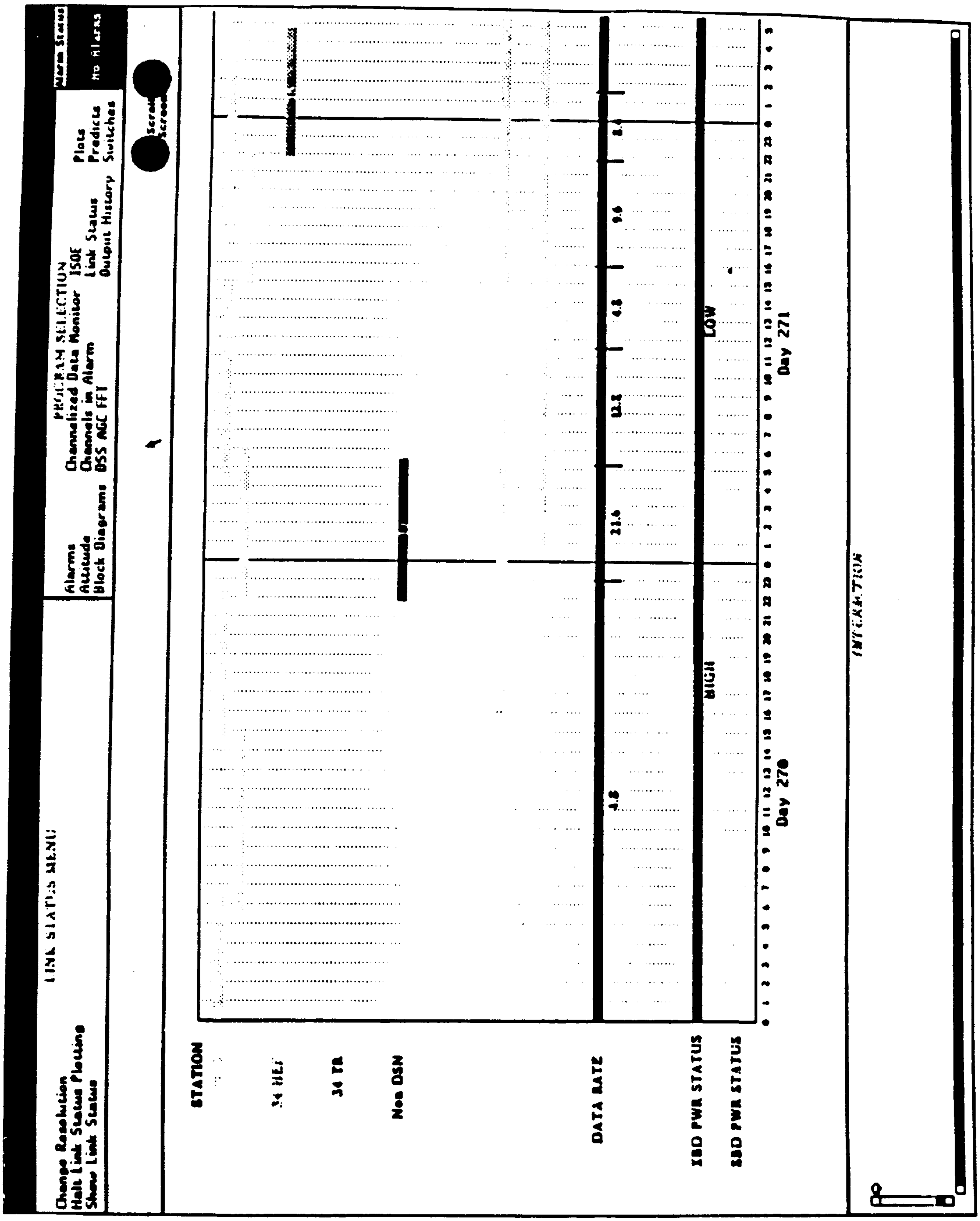


Figure 4-11. Link Status

4.8 Case Data Sources

Atkinson, D. J., D. L. Lawson, and M. L. James (July, 1989), "Artificial Intelligence for Multi-Mission Planetary Operations", *Proceedings of the Third Annual Workshop on Space Operations Automation and Robotics (SOAR).*

Lawson, D. L. and M. L. James, *SHARP: A Multi-Mission AI System for Spacecraft Telemetry Monitoring and Diagnosis.*

Malin, J. T., D. L.Schreckenghost, D. D. Woods, S. S. Potter, L. Johannesen, M. Holloway, and K. D. Forbus (September, 1991), *Making Intelligent Systems Team Players: Case Studies and Design Issues, Volume 1. Human-Computer Interaction Design*, NASA Technical Memo, Houston, TX: NASA - Johnson Space Center.

Martin, R. G. [Ed], D. J. Atkinson, M. L. James, D. L. Lawson, H. J. Porta (August, 1990), *A Report on SHARP and the Voyager Neptune Encounter*, JPL Publication 90-21, NASA.

Section 5
Knowledge-based Autonomous Test Engineer (KATE)

5.1 System Description

The KATE system provides autonomous diagnosis and control of complex electro-mechanical launch processing systems. It provides these functions using a model-based expert system, building a mathematical model of the system from information contained in the knowledge base. The performance of the model is compared to the measured performance of the system to detect anomalies, diagnose failures, and control the system.

The diagnoser analyzes the system and identifies the components which could cause the anomalous measurements. KATE evaluates the effects of the failure and takes the appropriate action, such as activation of redundant components or "safing" the system.

The user interface permits the operator to specify high or low level requirements to be met by the control system. These requirements are implemented by inferring control strategies from the description of structure and function contained in the knowledge base. Inconsistencies in the requirements are automatically identified by the control system. Additionally, manual control is possible.

Monitored Process

KATE is being applied to several domains, including Environmental Control System (ECS), Liquid Oxygen loading (LOX), and Water Tanking System (ALO-H2O). Figures 5-1, 5-2, and 5-3 provide the man-machine interface for these three applications. This discussion will focus on the scaled-down model of the Orbiter Maintenance and Refurbishment Facility's (OMRF) ECS. The ECS is designed to provide a conditioned flow of air to four different compartments of the orbiter while the Space Shuttle is being processed in the OMRF. This air flow is for the purposes of ventilation, cooling, and controlling static electricity discharge.

The ECS consists of a purge unit supplying chilled ventilation to four systems (payload bay duct, aft duct, fwd duct, and cabin access duct). Each of these systems contains a heater (for maintaining constant temperature) and a motorized flow-control valve (for controlling the flow rate).

Man-Machine System

Information about any component can be accessed by clicking on the component and making a selection from a menu. This functionality allows the user to set desired system requirements. For example, desired interface temperature (exiting air temperature) may be set for all four lines using this procedure.

Development and Testing Environment

KATE is implemented on a TI® LISP machine which is connected via a RS-232 port to the ECS test environment, a miniature version of the actual system. This provides real-time data acquisition capabilities. Malfunctions of various components can be introduced (for test and development purposes) on a control board connected to the system. A camera located in the ECS test building can be moved by KATE to view any component. This allows comparison of

---

® TI is a registered trademark of Texas Instruments Incorporated.

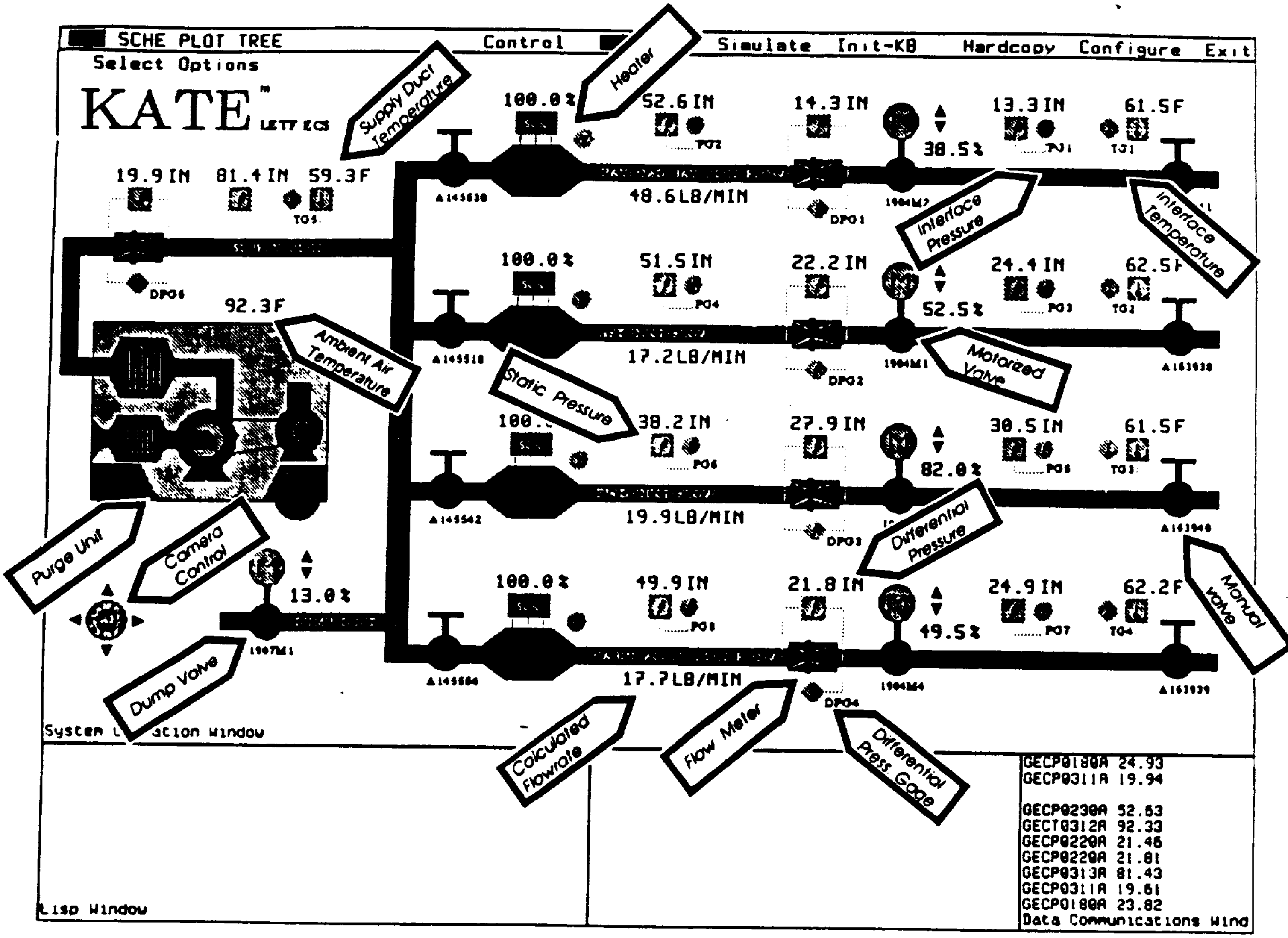


Figure 5-1. KATE Workspace for ECS Application

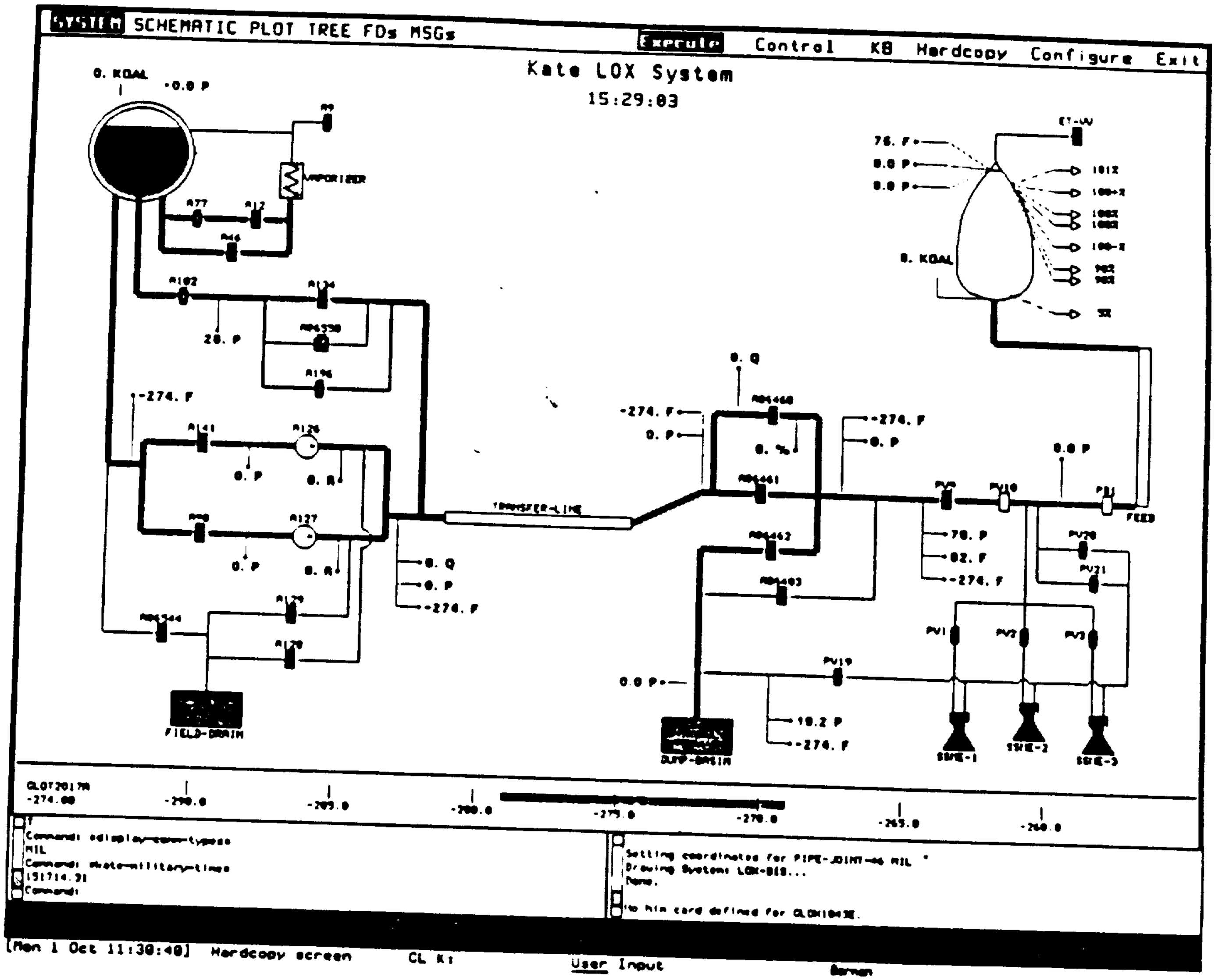


Figure 5-2. KATE Workspace for LOX Application

ORIGINAL PAGE IS
OF POOR QUALITY

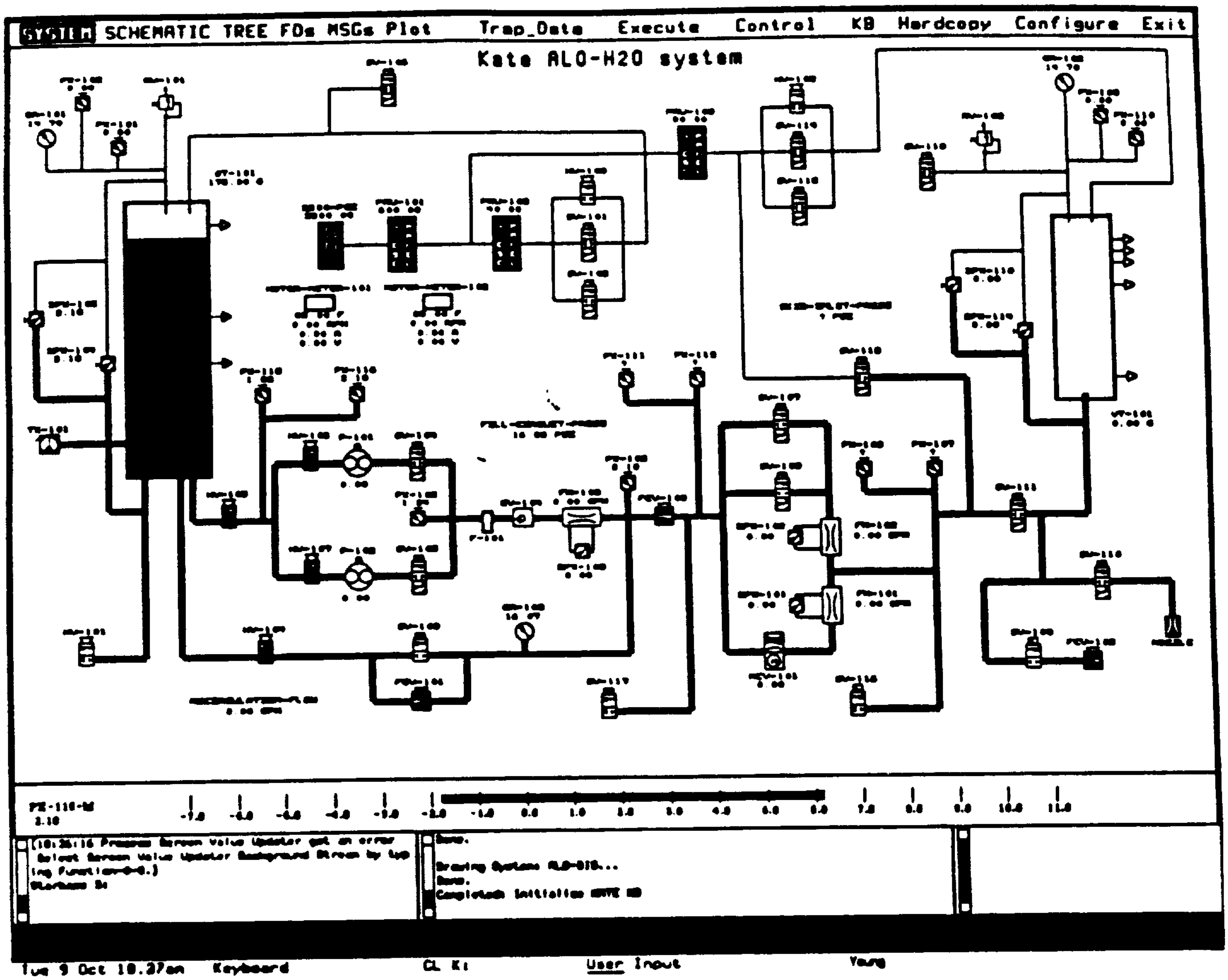


Figure 5-3. KATE Workspace for ALO-H20 Application

the actual component setting with the values indicated by KATE. This development/test environment is depicted in figure 5-4.

5.2 Intelligent System and Functions

KATE is a model-based system, describing the system of interest in terms of structure and function. It contains an internal software model of the ECS internalized to actual steady state system measurements. Expected measurement values are computed by taking command values sent to the system hardware and propagating them through the model. These expected values are then compared to the actual measurements from the ECS hardware. Discrepancies between these two values initiate a diagnosis to determine the fault.

Starting at the discrepant measurement, the diagnoser works backward through the model collecting all components which could affect the discrepant measurement and the set of all measurements from these components. The internal model is used to eliminate components based on their inability to produce the exact set of current measurements. If only one component remains, the diagnosis is complete. Otherwise, a list of culprits remains.

It should be noted that the KATE diagnosis only checks for single failures, not simultaneous, independent, multiple failures (the claim is that single-point failures is the norm).

5.3 Human-Intelligent System Interaction Functions

Assessment

1) System state

   Assessment of system state is accomplished by a schematic of the ECS which includes a graphic representation of the ECS, digital values of measurements for each of the system components, a LISP window, a command window, and an incoming data window (see figure 5-1).

2) Change in system state

   Since the value (flow rate, temperature, etc.) is indicated digitally on the schematic, there is a "data communications" window in the lower right portion of the display which is updated each time a sensor value changes (see figure 5-1). This is designed to aid in the presentation of direction and rate of change of data. However, this does require that the operator scan this window for the appropriate sensor, locate any updated measurements, compare these measurements, and compute direction and rate of change.

3) Diagnosis

   When a discrepancy (fault) occurs, the value for that component is indicated in red. After a diagnosis has been made, the color changes to purple. The operator may then take action to assess the nature of the fault. This is accomplished by clicking the mouse on the desired component (as mentioned previously). Selection from a menu presents one of several windows, including knowledge base frame, tree display of power bus, graph of limits and tradeoffs of flow, schematic of component, and historical data plot. From these windows information can be gathered as to the nature of the fault.

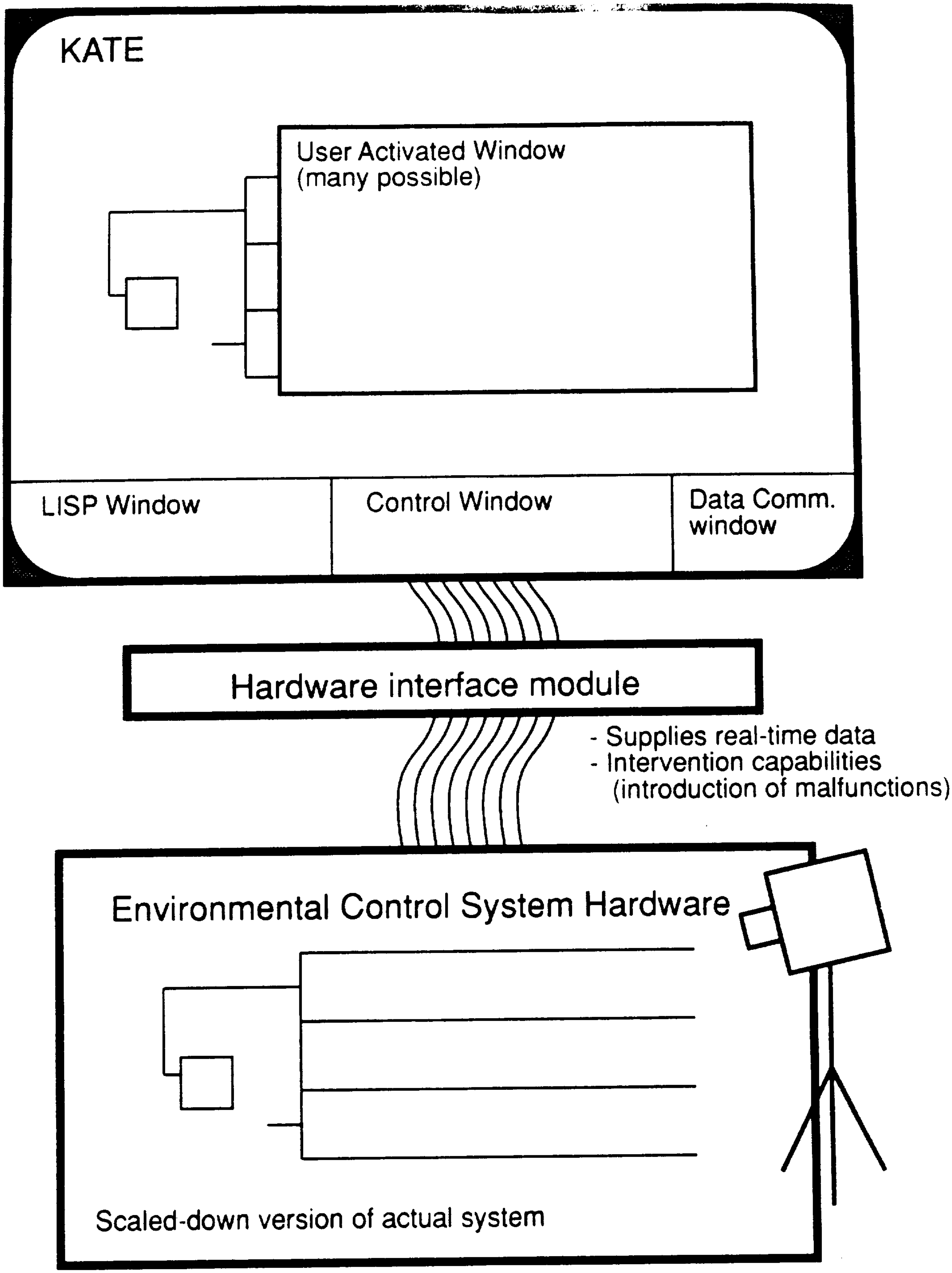


Figure 5-4. KATE Development and Test Environment

4) Inconsistent sensor reading

Determination of the cause of an inconsistent sensor reading will be described by an example. A case was described in which a sensor (valve) was indicating a value which was inconsistent with the model-based knowledge about the sensor. So, the value was indicated in purple. The operator attempted to understand what was the cause of this inconsistency. To do this, he clicked on the component and called up a graph used to illustrate limits and actual values for the sensors. At this point, a window similar to figure 5-5 was displayed. This graph indicated that the flow chosen by KATE was beyond the range of the particular sensor. Thus, the value was highlighted due to the fact that the sensor reading was offscale. As an aid in verifying this conclusion, the camera had moved to this particular component, displaying the gauge with the needle all the way to the right end of the scale.

Collaboration

1) Control

In order for the operator to see the control actions that KATE has performed, he/she must look in the command window (bottom center window in figure 5-1). In the present implementation, there is no ability to scroll up to control messages that have scrolled off the window. However, implementation of KATE on a different platform is planned which will provide scrollable windows. The results of the control sequences on component status are displayed by the digital values in the schematic display.

2) Diagnosis

Justification of diagnosis is provided in the LISP window in the form of information such as measurements used for the evaluation, statements concerning anomalous measurements, and remaining suspects in the diagnosis.

Intervention and Takeover

KATE operates in an autonomous role in which information is available only as feedback to the operator. However, the operator can manually change the setpoint of any component if warranted (and KATE will respond by adjusting the valves and heater output to achieve the setpoints).

5.4 Supporting User Interface Capabilities

Workspace

Figure 5-1 shows the organization of the workspace, with labels included to describe components. The largest window consists of a schematic of the system with digital data display for each of the components. Three smaller windows at the bottom of the workspace provide information about diagnosis and control from the expert system and changes in sensor readings. This implementation requires the operator to divide attention between the schematic diagram and the information windows.

Information and Presentation

The main window in KATE showing a physical mimic of the ECS is overlaid with additional windows as requested. As previously described, by clicking on a component, the operator can request a variety of information about that component. This selection activates an additional window.

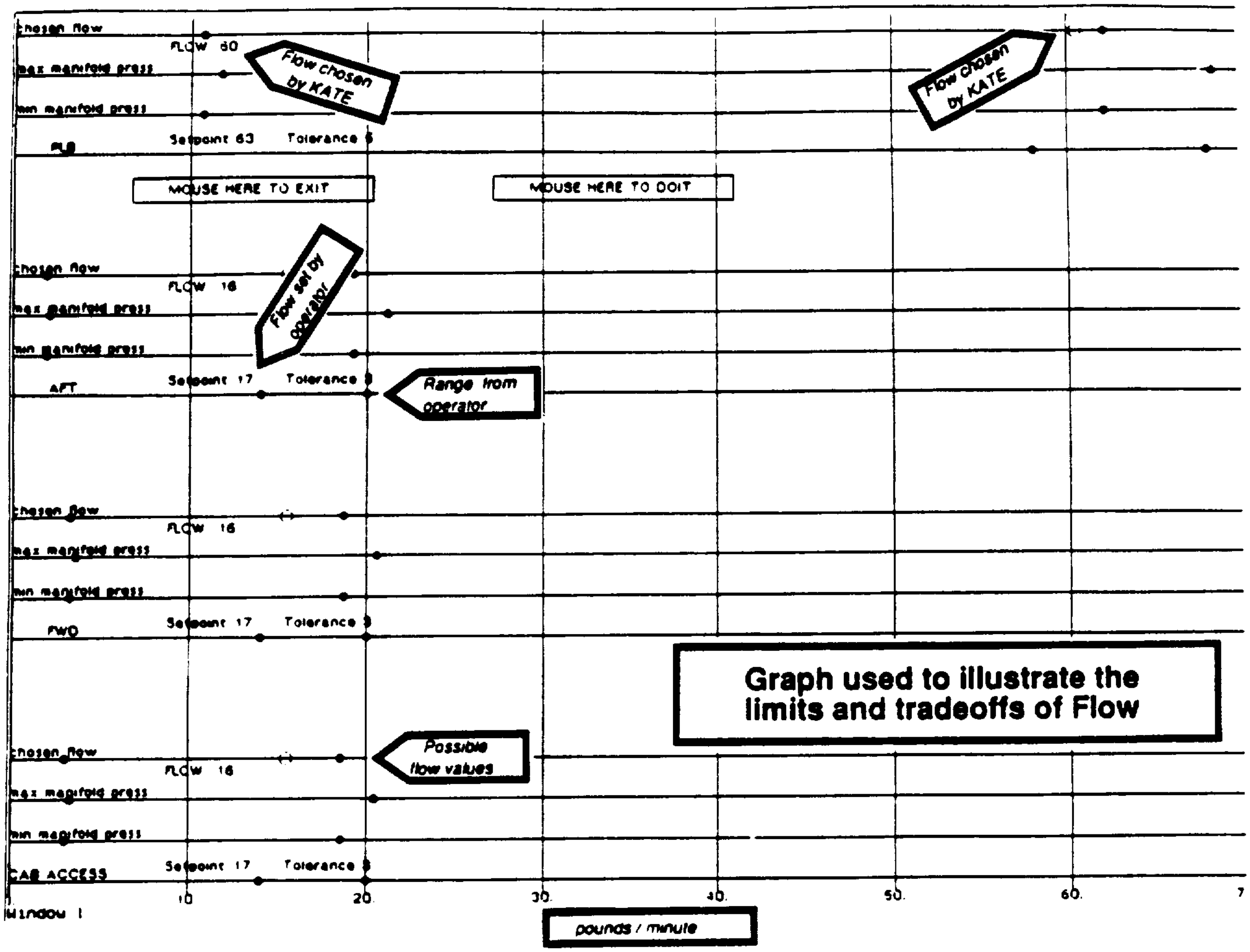


Figure 5-5. KATE Limit Graph

5.5 Summary of Issues

- Workspace navigation
  This case study is similar to others in that additional capabilities appear to have been added through additional windows, rather than exploring possibilities of integrating data to provide more information. This has the potential to create data management burdens and workspace navigation difficulties. See section 5.2 (Workspace: Proliferation of Windows) in Volume 1 (Malin et al, 1991) for more details on issues to be addressed.

- Physical topology schematic display
  High resolution graphic capabilities are used to provide a schematic display of the process. This schematic display is used primarily to show the interconnectivity of the components and to render individual components -- static portion of the display. State variables (that which the operator needs to be able to assess the state of the system) are displayed digitally, annotated on the schematic display. This approach may not adequately highlight events and anomalies in the process. To accomplish this, additional techniques may need to be used. The potential of analog forms that integrate related data is underutilized, as are additional techniques such as flow path coding, qualitative icons, etc. See section 5.4 (Physical Topology Schematic) in Volume 1 (Malin et al., 1991) for more information.

- Message Lists
  Intelligent system diagnosis and control actions are presented to the operator through a typical textual message list. As noted in section 5.3 (Message Lists and Timeline Displays) in Volume 1 (Malin et al., 1991), this approach may have difficulty in conveying temporal organization of sequence of events.

5.6 Study Method

Study Team

- David Woods (Ohio State University)
- Scott Potter (Ohio State University)

Project Representatives

- Jack Galliher (NASA KSC)
- Carrie Belton (NASA KSC)
- Barbara Brown (NASA KSC)
- Steve Beltz (Boeing)

5.7 Case Data Sources

Belton, C. L. and B. L. Brown, *KATE: A Model-based Control and Diagnostic Shell.*

Belton, C. L. and S. Enand, *KATE: A Model-based Diagnostic and Control Shell.*

Cornell, M. (July, 1987), *The KATE Shell: An Implementation of Model-based Control, Monitor, and Diagnosis*, pp. 355-360.

KSC, "KATE (Knowledge-based Autonomous Test Engineer) Environmental Control System Demonstration", presentation material.

Malin, J. T., D. L.Schreckenghost, D. D. Woods, S. S. Potter, L. Johannesen, M. Holloway, and K. D. Forbus (September, 1991), *Making Intelligent Systems Team Players: Case Studies and Design Issues, Volume 1. Human-Computer Interaction Design*, NASA Technical Memo, Houston, TX: NASA - Johnson Space Center.

Section 6
Intelligent Launch Decision Support System (ILDSS)

6.1 System Description

The ILDSS project is targeted with developing prototypes of tools for the launch team that support decision-making during Space Shuttle launch countdowns. ILDSS is composed of two parts: a Time Management System (TMS) and an Anomaly Management System (AMS). TMS is pointed at providing the NASA Test Director (NTD) with real-time support for the complex and changing relationships that exist between clock time, countdown time, events, and time windows during launch countdown. While the AMS is directed towards the Space Shuttle Project Engineer, some of the information in the AMS is directly useful to the NTD. These systems are developed in three phases: an information phase, a tool phase, and an advisor phase. Each phase consists of a concept prototype, a field prototype, and field testing. The following are the components of ILDSS:

- Time Management Integrated Display (TMID)
  TMID is the information phase. It integrates clocks and timelines with limited graphics and computes some decision-making aids. It is about to begin the field testing.

- Time Management What-If (TMWI)
  TMWI is the tool phase. It is a spreadsheet for doing what-if computations and uses constraint propagation. It is a field prototype.

- Time Management Situation Advisor (TMSA)
  TMSA is the advisor phase. It monitors terminal countdown and proactively suggests tactics for resuming the countdown when the countdown is not as planned. It is a concept prototype.

- Anomaly Management Launch Commit Criteria (AMLCC)
  AMLCC is the information phase. It provides on-line access to Launch Commit Criteria (LCC) during terminal count and automatically displays summary LCC information when an LCC violation occurs during terminal count.

- Anomaly Management Troubleshooting Tools (AMTT)
  AMTT is the tool phase. It provides on-line access to current precomputed products along with search. It is planned for a FY 92 start.

- Anomaly Management Troubleshooting Advisor (AMTA)
  AMTA is the advisor phase. It proactively identifies existing procedures or generic procedures for troubleshooting LCC violations. It is planned for a FY 92 start as well.

Our investigation focused on the TMID.

Monitored Process

The process of concern is the dynamic nature of the launch situation. As holds are imposed, for example, relationships and constraints change. These constraints require time-critical decisions to be made.

For example, there are:

- Dependencies between various steps,
- Time limits for transition between some events,
- Restrictions on scheduling certain events,
- Some events, such as T-0, that can be put on hold,
- Some events, such as Auxiliary Power Unit (APU) start that can be put on hold if they have not begun, but once started, are linked to clock time and cannot be extended,
- Some events, such as launch windows and collision avoidance over launch area (COLAs), which are tied only to clock time and thus cannot be put on hold.

Man-Machine System

TMID presents real-time data as events and changes occur and functions as an organized data display, to assist NASA Test Director (NTD) in visualizing and managing time relationships. Time management by the NTD consists of those decisions required by the extension of the T-9 minute hold or invocation of an unplanned hold at one of the remaining hold points. This requires maintaining mental models of the dynamic state of the countdown and managing the process that determines when to resume the count. TMID integrates information currently available on firing room clocks and hard-copy timelines and correlates the information graphically in real time.

Development and Testing Environment

The test environment involves access to real time Launch Processing System data external to the launch environment. This allows data collected from previous missions to be "played back" through the TMID. Additionally, real-time data acquisition is possible (and in fact critical) to exercise the visualization aspects of the system. Transfer of the field prototype into the launch environment is planned for 1991.

6.2 Intelligent System and Functions

From our perspective, TMID could be described as an intelligent interface rather than an intelligent system. This is due to the fact that the main function is the integration and presentation of the changing temporal relationships. However, TMSA will be intelligent.

6.3 Human-Intelligent System Interaction Functions

Assessment

1) System state

Assessment of system state is accomplished by the left window of the workspace (see workspace description in the following section). This provides, in graphic format:

- Launch window
- Countdown time
- Any scheduled holds
- COLAs
- APU status
- Status indication if count resumed (only during holds)

2) Change in system state

As system state changes, relationships between the Universal Time (UT) timeline and the Countdown Time (CDT) timelines change. Therefore, the coordination between these two timelines is the primary means that TMID presents changes in system state.

3) Diagnosis

There does not appear to be any requirement for diagnosis in TMID. TMSA will contain diagnosis.

Collaboration

As the system's function is that of data presentation, there is no provision or requirement for collaboration. TMS requires collaboration, however.

6.4 Supporting User Interface Capabilities

Workspace

The workspace is composed of multiple pages. Several pages are provided for entering, editing, and viewing data pertinent to the task (such and the number and timing of COLAs, etc.). The main page (presented in figure 6-1) is for information presentation and consists of vertical timelines, a horizontal "now" bar, clocks, and commands.

The left window is the UT timeline and presents information linked to clock time such as COLAs and any events that, once started, have a fixed duration.. This information is presented in three sub-windows. They are resume window, launch window, and APU runtime window. Each row represents 1 minute of UT and is labeled correspondingly.

The resume window (left part of UT timeline) uses green, yellow, and red color coding to convey (in traffic light fashion) the situation if the count were resumed at that point in UT (it is only active when there is a hold pending or when there is a hold). Coding depends on CDT, the launch window, COLAs, and APU run time. Yellow indicates lack of full APU contingency run time. Red indicates T-0 would either fall past the end of the launch window or within a COLA.

The launch window (center part of UT timeline) is linked solely to UT and indicates the presence of COLAs (in red on a blue background).

The APU runtime window (right part of UT timeline) is anchored in UT after the APUs are started. This indicates APU runtime limits. Prior to APU start, it depends on CDT and adjusts as a result of holds being invoked. The window changes to green and yellow after start, with green indicating nominal time to T-0 and yellow indicating APU contingency runtime.

The center window presents the CDT timeline consisting of information related to countdown time. Each row represents 1 minute of UT but is labeled in CDT. Holds are labeled in yellow with an H suffix. Ground Launch Sequencer milestones are shown in text in the minute for which they are planned. Planned events are indicated in white; cyan is used to represent events that have occurred.

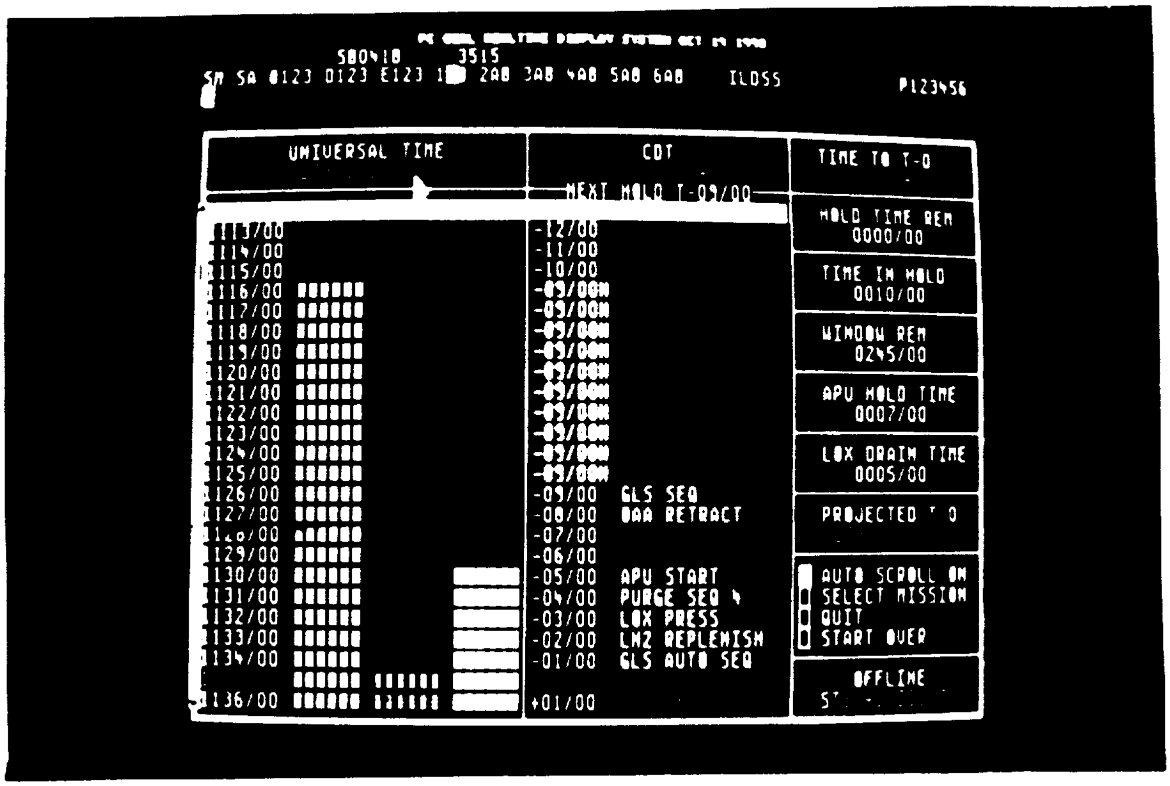


Figure 6-1a. ILDSS Workspace -- Mission #1 at Time 279:1112

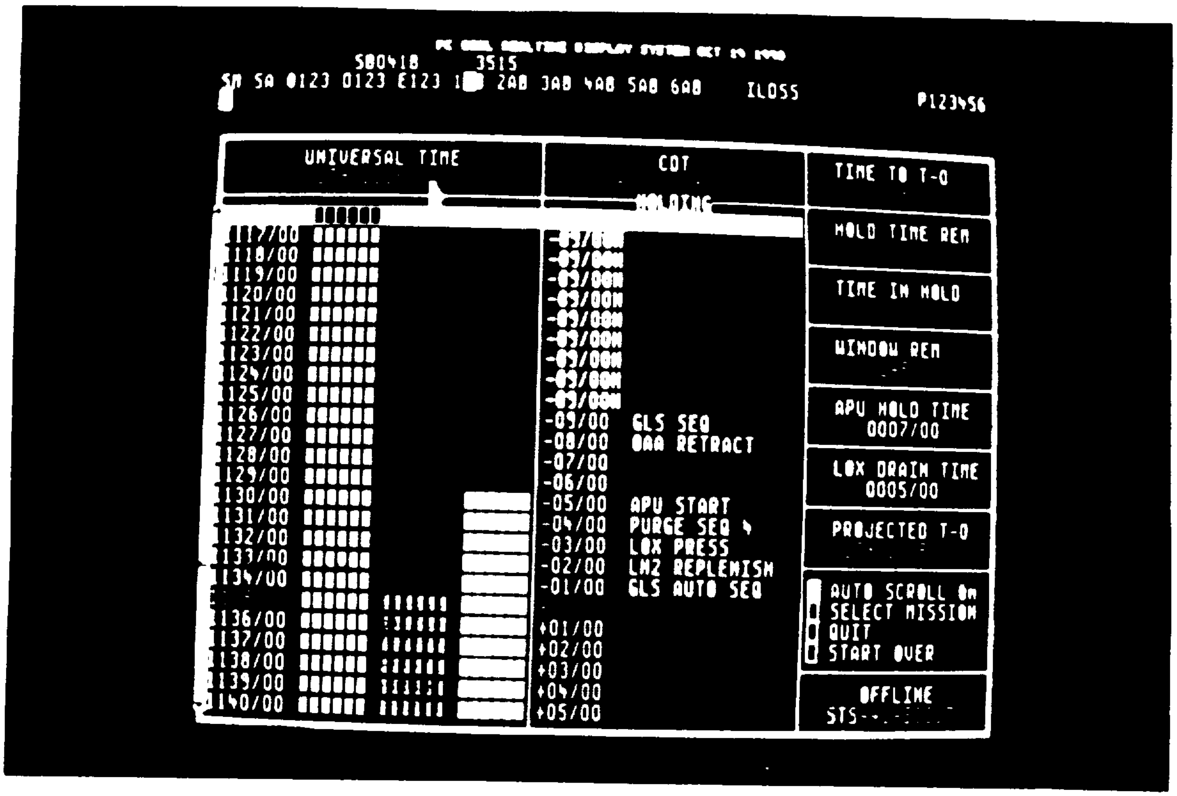


Figure 6-1b. ILDSS Workspace -- Mission #1 at Time 279:1116

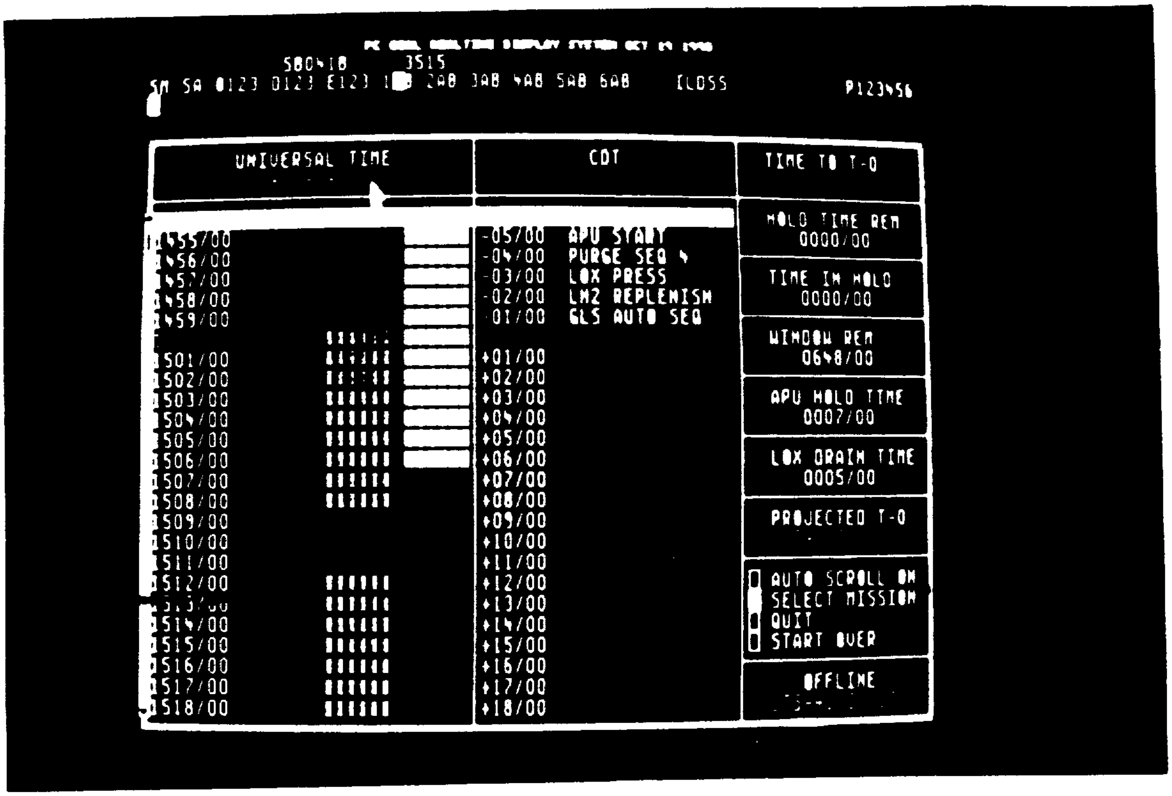


Figure 6-1c. ILDSS Workspace -- Mission #2 at Time 254:1454

Coordination of information between the UT and CDT timelines is accomplished by a "now" bar drawn horizontally across both timelines. The bar is positioned so that the current UT is within its boundaries. As time progresses, the bar is moved downward to the next row.

To demonstrate these capabilities, figure 6-1 indicates a scheduled hold at T-9 minutes. This figure exemplifies the relationships between these two timelines in two aspects. First, UT continues during a hold, so inserting a hold changes the alignment of events in the two timelines. Second, APU START is indicated at T-5 minutes on CDT, and also on UT, as once started, this event can only continue for 10 minutes. This window also provides messages related to current or upcoming system state (such as "NEXT HOLD T-09/00")

The right window provides digital values for seven additional clocks, selected based on interviews with NTDs. There are three types of clocks (with specifics in parentheses): time of day (UT and Projected T-0), contingency (Hold Time Remaining, Time in Hold, Window Remaining, APU Hold Time, and LOX Drain time), and countdown (time to T-0 and CDT). While not visible in the figure, pertinent parameters to the present system state are color-coded green, and non-pertinent parameters are coded blue.

Information and Presentation

TMID incorporates knowledge of time relationships so that the display can track any dependencies as they change.

Support for Interaction

Changes to the parameters can be entered by the operator by accessing the appropriate page in the display space. Figure 6-2 presents the mission profile page to provide an example of this type of interaction.

6.5 Summary of Issues

- Workspace

  Integration of Information. TMID demonstrates an integration of related time-management information, including UT, CDT, launch windows, planned events, and launch conflicts. These data are presented in parallel and the dynamic relationships are indicated to the user as changes occur. Workspace navigation problems and data management burdens (see section 5.2 in Volume 1, Malin et al., 1991) appear to be avoided. Additionally, color coding is used to indicate the relevance of the digital clocks. This helps the operator attend to relevant information.

- Timeline Displays

  Time management information is presented on two timeline displays -- UT and CDT. This is an example of plan-driven timeline displays. Temporal relationships are dynamically represented in this format and state of the launch process appears to be easily visible. This is an example of a concept which may be used to create alternatives to the textual message lists found in most of the case studies.

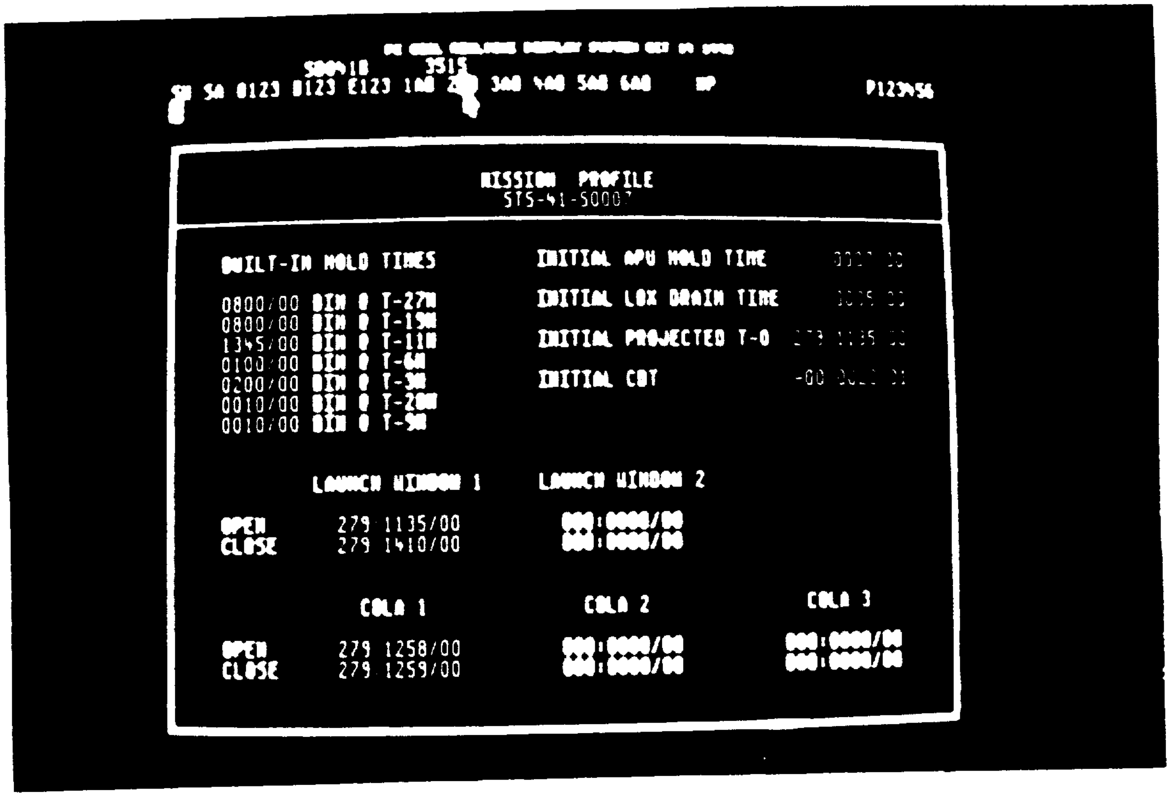


Figure 6-2. ILDSS Mission Profile Page

- Analog Graphic Forms

  TMID was implemented on PC hardware (IBM AT® compatible machines) to allow compatibility with existing Ground Operations Aerospace Language (GOAL) interface and character set software. This environment does not provide the availability of bit-mapped graphics as on a SUN® workstation. However, simple character-based graphics are elegantly used to indicate the important relationships across data (coordination of two timelines, timing and duration of events, contingencies, etc.).

6.6 Study Method

Study Team

- David Woods (Ohio State University)
- Scott Potter (Ohio State University)

Project Representatives

- Arthur Beller (NASA KSC)
- H. Greg Hadaller (Boeing Aerospace Operations)

6.7 Case Data Sources

Beller, A. E., H. G. Hadaller, and J. M. Ricci (October, 1990), *ILDSS Time Management Integrated Display PC Prototype Offline Short Manual*, Revision 1.0.

KSC (1990), *Research and Technology 1990 Annual Report*, NASA Technical Memorandum 103811.

Little, Arthur D. (August, 1988), *Intelligent Launch Decision Support System (ILDSS)*, final report to NASA KSC, Reference 55855.

Malin, J. T., D. L.Schreckenghost, D. D. Woods, S. S. Potter, L. Johannesen, M. Holloway, and K. D. Forbus (September, 1991), *Making Intelligent Systems Team Players: Case Studies and Design Issues, Volume 1. Human-Computer Interaction Design*, NASA Technical Memo, Houston, TX: NASA - Johnson Space Center.

® IBM AT is a registered trademark of International Business Machines, Inc.

® SUN is a registered trademark of Sun Microsystems Inc.

Appendix
Points of Contact for the Case Study

Space Shuttle Real-Time Data System (RTDS)
Johnson Space Center
Troy Heindel/NASA

Space Shuttle Guidance, Navigation and Control (GNC) Intelligent Systems
Johnson Space Center
Dave Miller/RSOC
Ron Montgomery/RSOC

Space Shuttle Instrumentation and Communications Officer (INCO) Expert System Project
Johnson Space Center
Art Rasmussen/MITRE

Space Shuttle KU Band Self Test Expert System
Johnson Space Center
George Pohle/RSOC

Space Shuttle DATA COMM Expert System
Johnson Space Center
George Pohle/RSOC

Space Shuttle Payload Deployment and Retrieval System Decision Support System (DESSY)
Johnson Space Center
Don Culp/RSOC
Dave Mayer/RSOC
Joe Watters/formerly RSOC
Kristen Farry/formerly RSOC
Gordon Johns/MITRE
Mark Gnabasik/MITRE
Mary Czerwinski/ formerly LESC
Benjamin Beberness/LESC

X-29 Remotely Augmented Vehicle Expert Systems (RAVES)
Ames Research Center Dryden Flight Research Facility
Dale Mackall
Dorothea Cohen

Military Aircraft Real-time Interactive Monitoring Systems (RTIMES)
Edwards Air Force Base
Robin Madison

Space Shuttle Onboard Navigation (ONAV) expert system
Johnson Space Center
Lui Wang/NASA
Malise Haynes/NASA

Space Shuttle Rendezvous Expert System
Johnson Space Center
Hal Hiers/NASA
Oscar Olszewski/LESC

Space Station Procedures Onboard Management System (OMS) Prototypes
Johnson Space Center
Christine Kelly/MITRE
C.Jayne Guyse/MITRE
Dave Hammen/MITRE
Chris Marsh/MITRE

Space Station Module/Power Management and Distribution (SSM/PMAD)
Marshall Space Flight Center (MSFC)
Louis Lollar/NASA Marshall Space Flight Center
Dave Weeks/NASA Marshall Space Flight Center
Bryan Walls/NASA Marshall Space Flight Center
Barry Ashworth/Martin Marietta
Michael Elges/Martin Marietta
Laura Jakstas/Martin Marietta
Chris Myers/Martin Marietta
Joel Riedesel/Martin Marietta

Space Shuttle Knowledge-Based Autonomous Test Engineer (KATE)
John F. Kennedy Space Center (KSC)
Jack Galliher/NASA Kennedy Space Center
Carrie Belton/NASA Kennedy Space Center
Barbara Brown/NASA Kennedy Space Center
Steve Beltz/Boeing Aerospace Company

Space Shuttle Intelligent Launch Decision Support System (ILDSS)
John F. Kennedy Space Center
Arthur Beller/NASA Kennedy Space Center
H. Greg Hadaller/Boeing Aerospace Operations

Glossary

ABE	Arm-Based Electronics
AGC	Automatic Gain Control Channel
AI	Artificial Intelligence
AMLCC	Anomaly Management Launch Commit Criteria
AMS	Anomaly Management System
AMTA	Anomaly Management Troubleshooting Advisor
AMTT	Anomaly Management Troubleshooting Tools
AOS	Acquisition of Signal
APU	Auxiliary Power Unit
ARC	Ames Research Center
BCD	Binary Coded Decimal
BFS	Backup Flight System
BITE	Built-In Test Equipment
CBR	Case-Based Reasoning
CCTV	Closed Circuit TeleVision
CDT	Countdown Time
CIU	Controller Interface Unit
CLIPS	C Language Integrated Production System
COA	Course of Action
COAS	Crew Optical Alignment Sight
COLA	Collision avoidance Over Launch Area
CPU	Central Processing Unit
CRT	Cathode Ray Tube
CSC	Computer Sciences Corporation
CSEL	Cognitive Systems Engineering Laboratory
D&C	Display and Control
DAP	Digital Auto Pilot
DDD	Digital Display Device
DESSY	Decision Support System
DFRF	Dryden Flight Research Facility
DMS	Data Management System
DOF	Degree of Freedom
DR	Diagnostic Reasoner
DSA	Decision Science Applications
DSS	Deep Space Station
EAFB	Edwards Air Force Base
ECS	Environmental Control System
ED	Engineering Directorate
EE	End Effector
ETC	End-to-end Test Capability
FA	Flight Aft
FCR	Flight Control Room
FDF	Flight Data File
FDIR	Fault Detection, Isolation, and Recovery
FELES	Front End Load Enable Scheduler
FF	Flight Forward

FFT	Fast Fourier Transform
FISA	Format Independent Storage Array
FPCS	Flight Path Control Set
FPS	Feet Per Second
FRAMES	Fault Recovery and Management Expert System
GC	Generic Controller
GMT	Greenwich Mean Time
GNC	Guidance, Navigation and Control
GPC	General Purpose Computer
HCI	Human-Computer Interaction
HITEX	Human Interface to the Thermal Expert System
HSTD	High Speed Trajectory Data
IESP	INCO Expert System Project
ILDSS	Intelligent Launch Decision Support System
IMU	Inertial Measurement Unit
INCO	Instrumentation and Communications Officer
ISA	Integrated Status Assessment
ISB	Intelligent System Branch
ISD	Information Systems Directorate
ISOE	Integrated Sequence of Events
ITF	Integrated Test Facility
JPL	Jet Propulsion Lab
JSC	Johnson Space Center
KATE	Knowledge-Based Autonomous Test Engineer
KSC	Kennedy Space Center
LAN	Local Area Network
LCC	Launch Commit Criteria
LESC	Lockheed Engineering and Science Corporation
LLP	Lowest Level Processor
LOA	Line of Approach
LOC	Loss of Control
LOS	Loss of Signal
LOX	Liquid Oxygen
LPLMS	Load Priority List Management System
LRU	Line Replaceable Unit
MC	Mid-Course
MCC	Mission Control Center
MCCU	Mission Control Center Upgrade
MCDS	Multifunction CRT Display System
MCIU	Manipulator Controller Interface Unit
MDM	Muxer-DeMuxer
MECO	Main Engine Cut Off
MED	Manual Entry Device
MER	Mission Evaluation Room
MET	Mission Elapsed Time
MOC	Mission Operations Computer
MOD	Mission Operations Directorate

MPM	Manipulator Positioning Mechanism
MPSR	Multi-Purpose Support Room
MRL	Manipulator Retention Latch
MSBLS	Microwave Scanning Beam Landing System
MSFC	Marshall Space Flight Center
MSID	Measurement Stimulus Identification
MSK	Manual Select Keyboard
MTK	Model ToolKit
NASA	National Aeronautics and Space Administration
NC	Nozzle Controller
NMI	Nautical Miles
NRT	Near Real Time
NTD	NASA Test Director
OAET	Office of Aeronautics, Explorations, and Technology
OMRF	Orbiter Maintenance and Refurbishment Facility
OMS	Orbital Maneuvering System
OMS	Operations Management System
ONAV	Onboard Navigation
OSU	Ohio State University
PASS	Primary Avionics System Software
PBI	Push Button Indicator
PCMMU	Pulse Code Modulation Master Unit
PDI	Payload Data Interleaver
PDRS	Payload Deployment and Retrieval System
PET	Phase-Elapsed Time
PI	Payload Interrogator
PI	Procedures Interpreter
PL	Payload
PLA	Power Lever Angle
PRLA	Payload Retention Latch Assembly
PROGRESS	PROpulsion monitorinG Real-time Expert SyStem
PSP	Payload Signal Processor
RAV	Remotely Augmented Vehicle
RAVES	Remotely Augmented Vehicle Expert System
RBI	Remote Bus Isolator
RCS	Reaction Control System
RET	Run-Elapsed Time
REX	Rendezvous Expert System
RMS	Remote Manipulator System
RPC	Remote Power Controller
RSOC	Rockwell Shuttle Operations Company
RTDS	Real Time Data System
RTIMES	Real-Time Interactive Monitoring Systems
RTOP	Research and Technology Operating Plan
Rx	Recovery Expert
SADP	Systems Autonomy Demonstration Project
SES	Space Shuttle Engineering Simulation
SFOC	Space Flight Operations Center
SFOS	Spacecraft Flight Operations Schedule

SHARP	Spacecraft Health Automated Reasoning Prototype
SPAN	Spacecraft Analysis
SSCC	Space Station Control Center
SSFP	Space Station Freedom Program
SSM/PMAD	Space Station Module/Power Management and Distribution
STK	Schematic ToolKit
STOL	Short Take Off and Landing
TACAN	Tactical Air Command and Navigation System
TAE	Transportable Application Environment
TDAS	Thermal Data Acquisition System
TDRSS	Tracking and Data Relay Satellite System
TEXSYS	Thermal Expert System
TGO	Time to Go
THC	Translational Hand Controller
TIG	Time of Ignition
TMID	Time Management Integrated Display
TMSA	Time Management Situation Advisor
TMWI	Time Management What-If
TMS	Time Management System
TRIAGE	Technique for Rapid Impact Analysis and Goal Evaluation
TVC	Thrust Vector Control
UT	Universal Time
VDU	Visual Display Unit
WEX	Workstation EXecutive
WOW	Weight-On-Wheels
XTK	Executive ToolKit

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
October 1991

3. REPORT TYPE AND DATES COVERED
Technical Memorandum

4. TITLE AND SUBTITLE
Making Intelligent Systems Team Players: Case Studies and Design Issues
Volume 2: Fault Management System Cases

5. FUNDING NUMBERS

6. AUTHOR(S)
Jane T. Malin (NASA), Debra L. Schreckenghost (MITRE), David D. Woods, Scott S. Potter, Leila Johannesen, Matthew Holloway (Ohio State University)

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Automation and Robotics Division
NASA Johnson Space Center
Houston, Texas 77058

8. PERFORMING ORGANIZATION REPORT NUMBER
S-652

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Washington, D.C. 20546-001

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
NASA TM 104738

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Unclassified/Unlimited
Subject Category 59

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Initial results are reported from a multi-year, interdisciplinary effort to provide guidance and asssistance for designers of intelligent systems and their user interfaces. The objective is to achieve more effective human-computer interaction (HCI) for systems with real-time management capabilities (i.e., process monitoring and control during anomalous situations). Intelligent fault management systems within NASA have been evaluated for insight into the design of systems with complex HCI. Preliminary results include (1) a description of real-time fault management in aerospace domains, (2) recommendations and examples for improving intelligent system design and user interface design, (3) identification of issues requiring further research, and (4) recommendations for a development methodology integrating HCI design into intelligent system design.

14. SUBJECT TERMS
Human-computer interaction, user interface, intelligent system, design guidance, development methodology, real-time fault management

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	Unlimited